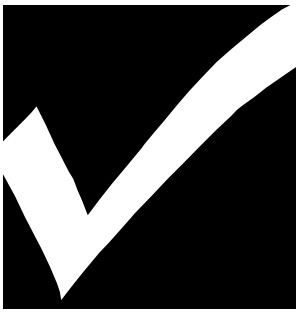
PEER REVIEW PROCESS

PR-PR-02 V3.0

NOVEMBER 20, 2002



Systems Engineering Process Office, Code 212
Space and Naval Warfare Systems Center San Diego
53560 Hull Street
San Diego CA 92152-5001

PREFACE

The purpose of Peer Reviews is to remove defects from work products early and efficiently. An important corollary is the development of a better understanding of the work products and of defects that might be prevented.

Peer Reviews involve a methodical examination of work products by the producer's peers to identify defects and areas where changes are needed. The specific products that will undergo a peer review are identified in the project's defined processes and scheduled as part of the project planning activities. Space and Naval Warfare (SPAWAR) Systems Center (SSC) San Diego Peer Review Policy states that projects at SSC San Diego shall utilize peer reviews in their project processes. This document provides guidance on planning and implementing Peer Reviews at SSC San Diego.

This Peer Review Process is available from the SSC San Diego Process Asset Library website at the following URL: http://sepo.spawar.navy.mil/

The SSC San Diego Systems Engineering Process Office (SEPO) assumes responsibility for this document and updates it, as required, to meet the needs of users within SSC San Diego. SEPO welcomes and solicits feedback from users of this document so that future revisions of this document will reflect improvements based on organizational experience and lessons learned. Please use the Document Change Request (DCR) form provided at the end of this document, or the online DCR form on the SSC San Diego Process Asset Library website (at URL: http://sepo.spawar.navy.mil/) to report deficiencies and/or corrections to the Peer Review Process document.

RECORD OF CHANGES

*__A__ - ADDED __M__ - MODIFIED __D__ - DELETED

VERSION NUMBER	DATE	NUMBER OF FIGURE, TABLE OR SECTION	__A*__ __M__ __D__	TITLE OR BRIEF DESCRIPTION	CHANGE REQUEST NUMBER
2.0	10/15/2000	Preface	A	Added to match HPM format	N/A
2.0	10/15/2000	Administrative Information	D	Deleted	N/A
2.0	10/15/2000	Throughout	M	Editorial and format changes throughout document to comply with HPM format	N/A
2.0	10/15/2000	Figures 2-2 & 2-3, Sections 2.5.1, 2.5.2 , 2.5.3 and 2.6	M	Add defect causal analysis to reviews as an optional step	PRP-0002 (partial)
2.0	10/15/2000	Appendix B	A	Focus area checklists	PRP-0004
2.0	10/15/2000	Throughout	A	Peer Review Data Collection Form	PRP-0004
2.0	12/14/2000	Sect. 1.2 and App A	A	Add PR KPA common feature traceability	PRP-0001
2.0	12/14/2000	Throughout	M	Make Peer Review Data Collection Form Mandatory	PRP-0004
3.0	11/20/2002	Throughout	M	Merge Peer Review and Formal Inspection Processes	PRP-0005

TABLE OF CONTENTS

Section **Page**

LIST OF FIGURES

Figure **Page**

LIST OF TABLES

Table **Page**

This page intentionally left blank.

SECTION 1. INTRODUCTION

1.1 PURPOSE

The purpose of this document is to provide Space and Naval Warfare (SPAWAR) Systems Center (SSC) San Diego projects with a process for planning and implementing peer reviews on work products and the processes used to develop or manage work products.

Implementing peer reviews on work products is recognized as a "Best Practice" in project management. Projects that conduct peer reviews as described in this process are in compliance with the SSC San Diego Peer Review Policy.

To simplify the discussion of applying peer reviews on products and processes, the word "product" is used to include both work products and the processes used to develop or manage work products.

The objective of peer reviews is to remove defects from the products early and efficiently. The results of the peer review are recorded and passed on to the author or person responsible for correcting defects discovered during the review.

1.2 BACKGROUND

Peer reviews have been used to detect and remove defects from work products early in the development cycle since the early 1990s. Early removal of defects results in a 100 to 1 reduction in cost of rework, when compared to rework cost for defects discovered in the testing phase or after product delivery. Another advantage is the development of a better understanding of the work products and of defects that might be prevented.

Peer Reviews are a product development best practice described in both the Capability Maturity Model for Software (SW-CMM), and the Capability Maturity Module Integrated (CMMI). In the CMMI, Peer Reviews are included as a Specific Goal of the Verification Process Area. In the SW-CMM, Peer Reviews is a Level 3 Key Process Area (KPA). Appendix A indicates in which section of this document each of the SW- CMM Peer Review KPA common features are addressed.

1.3 SCOPE

Peer reviews are an integral component of the SSC San Diego standard engineering process and should be conducted by all projects at SSC San Diego. This document describes three types of peer reviews: (1) Walkthroughs, (2) Technical Reviews, and (3) Formal Inspections (FI).

1.4 DEFINITIONS

Definitions of terms used throughout this document are listed below:

a. Peer Review - A peer review is a methodical examination of a product by the author's peers to identify defects and areas where changes are needed. The specific products that will undergo peer reviews are identified in the project's plans and scheduled as part of the project planning activities. Data concerning numbers of defects and types are collected.

b. Walkthrough - The objective of a walkthrough is to informally evaluate a product. Walkthroughs have been long associated with reviewing source code, but a walkthrough may also be applicable to other products such as architectural designs, detailed designs, test plans and procedures, and change control procedures. The major objective is to find defects, omissions, and contradictions;

to improve the product; and to consider alternative implementations. Other important objectives of the walkthrough include exchange of techniques, style variations, and education of the participants. A walkthrough may point out efficiency and readability issues in the code, modularity issues in design specifications or requirements testability issues. The walkthrough is the least formal of the three types of peer reviews discussed in this document. The author of the product and one or more reviewers conducts a walkthrough. Defect data is systematically collected and stored in a peer review database.

c. Technical Review - A technical review is a coordinated team evaluation of a product. It identifies deviations from specifications and standards, identifies defects, and may examine alternative solutions. The review team may provide recommendations for correction of defects and deviations. The technical review is more structured than a walkthrough yet is much less structured and informal than the Formal Inspection (FI). The technical review participants include the author, and a team of participants knowledgeable of the technical content of the product being reviewed. Defect resolution is mandatory, and rework may be formally verified by another Technical Review or informally reviewed. Defect data is systematically collected and stored in a peer review database.

d. Formal Inspection – The FI is a structured formal peer review. The purpose of an FI is to identify and classify product defects. The product author's peers conduct FIs. An FI team typically has three to six members. A moderator leads the process. Defect resolution is mandatory, and rework may be formally verified by re-inspection or informally reviewed. Defect data is systematically collected and stored in a peer review database.

e. Product - A deliverable or non-deliverable item produced or acquired during software development or maintenance. Some types and examples of products include requirements documents (System/Subsystem Specification, Software Requirements Specification), design documents, (Software Design Description, Database Design Description), source code, Test Plans, Test Procedures, User/Operator Manuals, Support Manuals (Firmware Support Manual), planning documents (Software Development Plan, Configuration Management (CM) Plan, Software Quality Assurance Plan) and process documents (CM Process, Peer Review Process).

f. Measure - A quantitative assessment of the degree to which a product or process possesses a given attribute.

g. Peer - A peer is an individual who is assigned to perform a peer review of the product. A peer is responsible for performing his or her role in accordance with the specific peer review chosen. The peer has a level of development expertise and product knowledge sufficient to comprehend the product under review. Peers are also referred to in this document as "inspectors", "reviewers", and/or "team members."

h. Author - The author is responsible for the product requiring a peer review and presents the material to the peer review team.

i. Process Goal – The objective of a process.

j. Roles and Responsibilities – The tasks and responsibilities of individuals or groups for accomplishing the process.

k. Entry Criteria – The elements and conditions necessary to be in place to begin the process.

l. Input – Data and/or material on or with which a process is performed.

m. Procedure – The actions to transform an input, as influenced by controls, into a predetermined output.

n. Output – Data and/or material, produced by or resulting from a process. It must include the input in some form.

o. Exit Criteria – Elements and/or conditions necessary to be in place to complete a process.

p. Focus Area – Particular area of concern on which a reviewer is assigned to focus.

q. Major Defect – A significant error or omission in a work product that will result in a malfunction or unexpected outcome if uncorrected. Major defects are considered the highest priority problems in software development.

r. Minor Defect – An error or omission that does not cause or lead to a malfunction. Minor defects are considered low priority problems in software development.

s. Open Issue – A problem not easily classified as a major/minor defect, or any defect under discussion for more than two minutes during a review meeting which is subsequently deferred for resolution.

t. Redline Errors – Spelling, punctuation, grammar, syntax noted for corrective action. Redline errors are not considered defects or open issues.

1.5 ABBREVIATIONS AND ACRONYMS

CM	Configuration Management
CMMI	Capability Maturity Model Integrated
DCR	Document Change Request
DID	Data Item Description
FI	Formal Inspection
IEEE	Institute of Electrical and Electronics Engineers
IRS	Interface Requirements Specification
KPA	Key Process Area
LPH	Lines of code Per Hour
MIL	Military
NAVSSI	Navigation Sensor System Interface
PAL	Process Asset Library
PDL	Program Design Language
PPH	Pages Per Hour
SDF	Software Development File or Folder
SDP	Software Development Plan
SEPO	Systems Engineering Process Office
SPAWARSYSCOM	Space and Naval Warfare Systems Command

SPS	Software Product Specification
SQA	Software Quality Assurance
SRS	Software Requirement Specification
SSC	Space and Naval Warfare Systems Center
STD	Standard
SVD	Software Version Description
S/W	Software
SW-CMM	Software Capability Maturity Model
VDD	Version Description Document

1.6 TAILORING GUIDELINES

The procedural steps established here define the SSC San Diego organizational process for peer reviews. Projects should tailor this process as appropriate to the size and scope of the project and product being reviewed.

1.7 DOCUMENT OVERVIEW

Section 1 is an introduction to the Peer Review Process. Section 2 is a description of the process. Section 3 contains information to support the conduct of peer reviews. Appendix A is a SW-CMM Peer Review KPA requirements matrix. Appendix B provides product checklists to be used in reviewing software work products. Appendix C provides forms and checklists to be used when conducting peer reviews.

1.8 REFERENCE DOCUMENTS

The following documents were used to create or are referenced in this process:

a. SSC San Diego Software Engineering Process Policy – SPAWARSYSCENINST 5234.1

b. SSC San Diego Peer Review Policy

c. FI Log Form Merge Procedure

d. FI Log Form Merge Procedure (Expert Mode)

e. Institute of Electrical and Electronics Engineers (IEEE) Standard for Software Reviews and Audits, IEEE Std 1028-1988

f. Formal Inspection Procedure (Expert Mode)

g. Peer Review Procedures for NCCOSC RDTE D87 Software Work Products

h. Software Inspection, Tom Gilb and Dorothy Graham, Addison-Wesley, 1993

i. Technical Review Procedure (Expert Mode)

j. Walkthrough Procedure (Expert Mode)

SECTION 2. PROCESS DESCRIPTION

2.1 PEER REVIEW PROCESS OVERVIEW

Figure 2-1 is an overview of the four phases of the Peer Review Process. The phases are summarized below:

a. Planning Phase – The following planning activities are conducted:

   1. Roles and responsibilities are assigned to meet peer review objectives
   2. Entrance criteria for conducting the peer review are addressed
   3. The type of peer review appropriate for the product is selected

b. Implementation Phase involves scheduling, preparing for, and executing peer review meetings.

c. Measurements and Data Collection is an activity that is performed to assess cost and benefit effectiveness of implementing Peer Reviews.

d. Peer Review Audit is a Software Quality Assurance (SQA) activity that is performed to audit the actual implementation of the project's Peer Review Process.

2.2 UNDERSTAND ROLES AND RESPONSIBILITIES

The roles and responsibilities described in the following sections apply regardless of the type of peer review. Specific responsibilities for participants in a walkthrough, technical review, and formal inspection are discussed in subsequent paragraphs.

2.2.1 Project Management

Project management is responsible for the following activities:

a. Approving the Peer Review Policy.

b. Ensuring that the project Software Development Plan (SDP) or other appropriate planning document includes peer reviews and specifies the type of peer review for each product.

c. Providing the necessary resources of time, personnel, budget, and facilities required to plan, define, execute, and manage reviews.

d. Ensuring that reviews are conducted, the results of a peer review are reviewed against project milestone events, and that product rework is accomplished.

e. Requiring training and orientation for the project staff in the use of the Peer Review Process.

2.2.2 Project Staff

The project staff is responsible for implementing the Peer Review Policy and attending training and orientation in the use of the Peer Review Process. It is essential that participants selected for a particular peer review have a level of development expertise and product knowledge sufficient to comprehend the product under review.

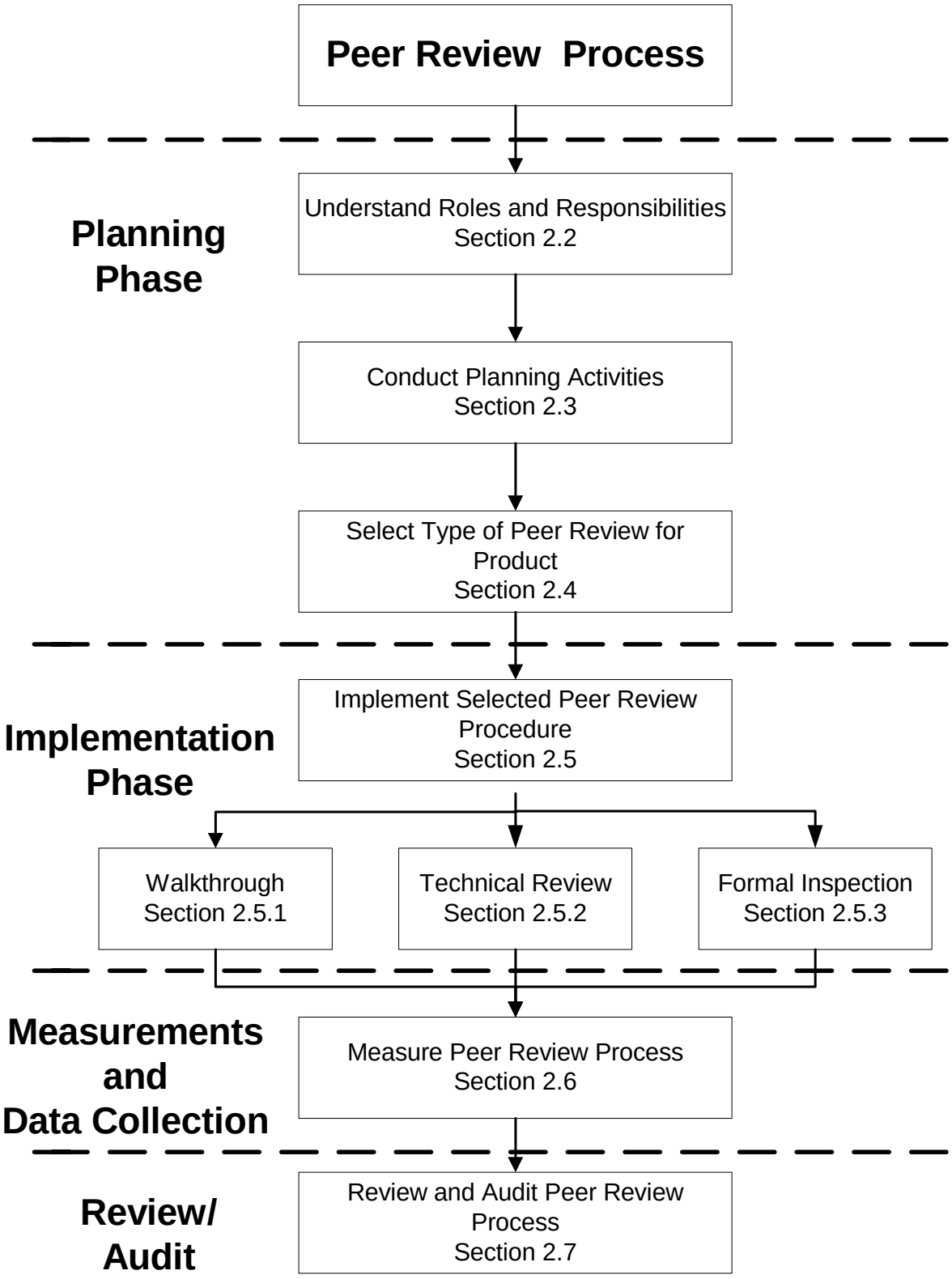


Figure 2-1. Peer Review Process Overview

2.2.3 Author

The author is responsible for the product requiring a peer review and presents the material to the peer review team. In the case of an FI, a designated individual other than the author presents the material to the FI team. In all cases, the author is responsible for resolving defects and open issues.

2.2.4 Peer

A peer is an individual who is assigned to perform a peer review of the product. A peer is responsible for performing his or her role in accordance with the specific type of peer review. The peer should have a level of development expertise and product knowledge sufficient to comprehend the product under review. Peers are also referred to in this document as "inspectors" and/or "team members."

2.2.5 Software Quality Assurance

SQA provides the independence in reviewing the implementation of the Peer Review Process. SQA ensures that the type of peer review is appropriate for the product. SQA ensures that discovered defects are corrected and open issues are resolved. Results are reported to project management.

2.2.6 Systems Engineering Process Office

The SSC San Diego Systems Engineering Process Office (SEPO) maintains this process document and updates it as required to meet the needs of users within SSC San Diego. This process document will be updated based on user input to reflect improvements, based on organizational experience and lessons learned. SEPO provides assistance to projects with the Peer Review Process and teaches a peer review training workshop. See the Upcoming Events page of the SSC San Diego Process Asset Library (PAL) at http://sepo.spawar.navy.mil for additional information on the Peer Review Workshop. Additionally, SEPO maintains the Peer Review database. SEPO receives defect logs and the data is entered into the Peer Review database. The data is used to compare the relative effectiveness of the Peer Review Process and for determining trends that may improve the process for participating projects, e.g., the average number of issues found during each inspection.

2.3 CONDUCT PLANNING ACTIVITIES

For any of the peer reviews, the planning activities listed below need to be conducted:

a. Product criteria are defined. Product criteria such as readability, modularity, consistency, functionality, testability, etc., are sufficiently defined. Software development and documentation standards have been identified. Software development and documentation standards include Military Standard (MIL-STD)-498, Data Item Descriptions (DIDs), and commercial standards such as various IEEE standards. The product criteria, development and documentation standards should be referenced in the project plans.

b. Checklists are defined. Checklists that further define product criteria and focus areas have been developed, and reviewed by the author's peers and potential checklist users. Use of checklists by reviewers assists in detecting and classifying potential product defects. Examples of software engineering-related work product checklists are found in Appendix B of this document.

c. The products to be reviewed are defined. The project development plan such as the SDP, Software Quality Assurance Plan, Software Configuration Management Plan, or Project Management Plan should define the products to be reviewed.

d. Peer Reviews are planned. Peer reviews are planned and scheduled as defined in the project plans. Peer Reviews are conducted to achieve the milestone dates set forth in the project plan.

e. Personnel are trained. Reviewers who participate in peer reviews receive required training in the objectives, principles, and methods of peer reviews. Additionally, the designated leaders of the peer reviews have received training on how to lead peer reviews.

f. Resources and funding have been identified and planned. Resources and funding have been identified, planned, and documented in the project plan for each product to be peer reviewed.

g. Mechanism is in place for peer review measurements. Data on the conduct and results of peer reviews are collected, maintained, and reported. Measurements are established to determine the status of the peer review activities. The Peer Review Announcement and Report (Form 1) in Appendix C provide examples of the type of data that is maintained in a Peer Review database.

2.4 SELECT TYPE OF PEER REVIEW FOR PRODUCT

Table 2-1 is an example of a Peer Review Selection Matrix. The column entitled "Product" shows the products to undergo a peer review. The column entitled "Technical Drivers" is the criteria used to determine the type of peer review to be used. In this example, "Complexity" is used as the technical driver and scaled as Low, Average and High. Complexity is defined as the degree to which a system or component has a design or implementation that is difficult to understand and verify. "Low Complexity" is defined as simple or easily verified. "Average Complexity" is defined as moderate and can be verified. "High Complexity" is defined as significant or difficult to verify. The foregoing and Table 2-1 are intended as an example only. The criteria for selection of the type of peer review and the products to be reviewed should be documented in project plans.

2.5 IMPLEMENT SELECTED PEER REVIEW PROCEDURES

The following sections provide the procedural descriptions for a walkthrough, technical review, and formal inspection. The procedural descriptions establish the minimum template used to plan, prepare, and execute any of these peer reviews.

2.5.1 Walkthrough Procedure

Figure 2-2 provides an overview of the walkthrough. See the expert mode of the Walkthrough Procedure, reference j, on the SSC San Diego PAL at http://sepo.spawar.navy.mil/ for an easy to use guide to performing walkthroughs.

2.5.1.1 Procedure Goal. The purpose of a walkthrough is to find defects, omissions, and contradictions to improve the product and to consider alternative implementations.

2.5.1.2 Special Responsibilities. The following roles are established for the walkthrough:

a. Author. The author is the person responsible for the product being examined, selects the peers to participate in the review, and presents the product.

b. Walkthrough Team. A walkthrough may have one or more reviewers. Each member is responsible for reviewing any input material prior to the walkthrough meeting, if held, and participating during the walkthrough meeting to ensure that it meets its objective.

c. Recorder. The author may request or assign a team member to perform the duties of a recorder. If assigned, the recorder is responsible for writing all comments made during the walkthrough meeting that pertain to errors found, questions of style, omissions, contradictions, suggestions for improvement, or alternative approaches.

TABLE 2-1. EXAMPLE OF A PEER REVIEW SELECTION MATRIX

Product	Technical Drivers - Complexity		
	Low	**Average**	**High**
Software Requirements	Formal Inspection	Formal Inspection	Formal Inspection
Design	Technical Review	Formal Inspection	Formal Inspection
Software Code and Unit Test	Walkthrough	Technical Review	Formal Inspection
Qualification Test	Technical Review	Technical Review	Formal Inspection
User/Operator Manuals	Walkthrough	Technical Review	Formal Inspection
Support Manuals	Walkthrough	Technical Review	Formal Inspection
Software Documents, e.g. Version Description Document (VDD), Software Product Specification (SPS), Software Version Description (SVD)	Walkthrough	Walkthrough	Walkthrough
Planning Documents	Formal Inspection	Formal Inspection	Formal Inspection
Process Documents	Technical Review	Formal Inspection	Formal Inspection

2.5.1.3 Entry Criteria. The entry criteria for a walkthrough are listed below:

a. The product is ready for a walkthrough

b. The applicable standards, guidelines, and checklists are available

c. Source documents from which the product is derived are available.

2.5.1.4 Input. The input to the walkthrough is the product to be reviewed.

2.5.1.5 Tasks. The tasks necessary to complete the walkthrough are described in the steps that follow.

Step 1. Plan the Walkthrough Meeting

The author completes the following activities:

a. Identifies the walkthrough team by selecting one or more people. If a recorder is needed, the author assigns this role.

b. Schedules the walkthrough meeting, if one is to be held, and selects the meeting place.

Procedure Goal Defined

Roles & Responsibilities Established

Entry Criteria Met
- Product is Ready
- Applicable Standards and Checklists are Available
- Source Documents are Available

Inputs
- Product subject to Walkthrough

Tasks
- Step 1: Plan the Walkthrough Meeting
- Step 2: Review the Product
- Step 3: Conduct the Walkthrough Meeting
- Step 4: Resolve Defects
- Step 5: Conduct Defect Causal Analysis
- Step 6: Resolve Open Issues
- Step 7: Document the Walkthrough
- Step 8: Rework the Product
- Step 9: Review & Audit
- Step 10: Follow-up

Output
- Walkthrough Documentation
- Reworked Product

Exit Criteria Met
- Product Rework Completed
- Walkthrough Documented

Figure 2-2. Walkthrough Procedure for Work Products

c. Distributes to the reviewers the product and all necessary material, such as checklists or standards to facilitate the review of the product. The author determines if an overview of the product needs to be given to educate the reviewers of the product. The author allows enough time for the reviewers to review the material.

Step 2. Review the Product

Reviewers are responsible for preparing for the walkthrough meeting by becoming thoroughly familiar with standards, checklists and any other information that was provided. Depending on the product being reviewed and whether a walkthrough meeting is to be held, reviewer(s) may actually review the product and prepare to discuss their comments, recommendations, questions, and redlines on the product prior to the walkthrough meeting. Often however, the review of the product occurs during the walkthrough meeting.

Step 3. Conduct the Walkthrough Meeting

The author walks through the product. The team member(s) may ask questions or raise issues on the product, and/or document their concerns. If a recorder is assigned, the recorder writes comments and decisions for inclusion in the walkthrough report and completes the Peer Review Announcement and Report (Form 1). See Appendix C for the Peer Review Announcement and Report (Form 1). If a recorder is not assigned, the author gathers this information.

During the walkthrough meeting, the reviewer(s) may recommend another walkthrough be performed if a large number of issues or defects were discovered.

Step 4. Resolve Defects

The author and the reviewer(s) resolve defects discovered in the walkthrough. Open issues may be referred to the project lead for resolution.

Step 5. Conduct Defect Causal Analysis (optional)

The author and reviewer(s) review defects discovered in the walkthrough to determine the probable cause and to recommend steps to prevent similar defects from being injected in the future. If a defect was caused by an error in another baselined product, follow the appropriate configuration management process to correct that product.

Step 6. Resolve Open Issues

The author meets with team members to resolve open issues and reach closure.

Step 7. Document the Walkthrough

The author documents that a walkthrough was conducted on the product. At a minimum, this should include identification of the product reviewed, the names of the reviewer(s), the date of the walkthrough, an overview or summary of deficiencies, omissions, contradictions, and suggestions for improvement. There are several appropriate places to document that a walkthrough was conducted such as a Software Engineering Notebook, or Software Development File or Folder (SDF). The Peer Review Announcement and Report (Form 1) is used to summarize the types and numbers of defects and effort expended for the walkthrough. Causal analysis findings should be documented if a causal analysis was conducted. Ownership of action items and status should also be documented. The Peer Review Announcement and Report (Form 1) should be filed in the SDF and a copy forwarded to SEPO.

Step 8. Rework the Product

The author reworks the product as recorded during the walkthrough.

Step 9. Review and Audit

SQA audits implementation of the process and all resulting artifacts.

Step 10. Follow-up

Author ensures open issues and action items are tracked to closure.

2.5.1.6 Output. The outputs of the walkthrough are listed below:

a. Documentation that a walkthrough was conducted (including Peer Review Announcement and Report (Form 1))

b. The reworked product.

2.5.1.7 Exit Criteria. The exit criteria for the walkthrough are listed below:

a. Product rework and second walkthrough (if required) is complete.

b. A summary of deficiencies, omissions, efficiency issues, results of causal analysis (if done), and suggestions for improvement has been produced.

c. The author documents that a walkthrough has been conducted in the appropriate notebook, file, or folder and completes the Peer Review Announcement and Report (Form 1).

d. Completed Peer Review Announcement and Report (Form 1) has been forwarded to SEPO.

2.5.2 Technical Review Procedure

Figure 2-3 provides an overview of the technical review. An expert mode of the Technical Review Procedure, reference i, is available on the SSC San Diego PAL.

2.5.2.1 Procedure Goal. The objective of a technical review is to evaluate a specific product and provide management with evidence of the following items:

a. The product conforms to the project's plans, standards, guidelines, and requirements.

b. Changes to the product are properly implemented, documented, and affect only those system areas identified by the change.

2.5.2.2 Special Responsibilities. The roles established for the technical review are listed below:

a. Review Leader. The author or creator of the product under review should be assigned as the Review Leader, if possible. In any case, the author or creator is required to be present at the review meeting. The review leader is responsible for conducting the technical review. This includes administrative tasks pertaining to the review and ensuring that the review is conducted in an orderly manner. The review leader is also responsible for distributing the Peer Review Announcement and Report (Form 1), and documenting action items resulting from the technical review. If an action item database is in place, the Review Leader is responsible for entering action items in the database. The Review Leader should track and monitor action items to closure.

b. Recorder. The recorder is responsible for documenting findings (e.g., defects, inconsistencies, omissions, and ambiguities), decisions, and recommendations made by the review team and completing the Peer Review Announcement and Report (Form 1).

c. Technical Review Team Member. Each team member is responsible for preparing for the review and ensuring the review meets its objectives. Together, team members are responsible for formulating recommendations in such a way that management can act on them promptly.

2.5.2.3 Entry Criteria. The entry criteria for a technical review are listed below:

a. The product is ready for a technical review

b. The project's plans, standards, guidelines, and checklists are available

c. Source documents from which the product is derived are available.

2.5.2.4 Input. The input to the technical review is the product to be reviewed.

Procedure Goal Defined

Roles & Responsibilities Established

Entry Criteria Met
- Product is Ready
- Applicable Plans, Standards and Checklists are available
- Source Documents are available

Inputs
- Product to be reviewed

Tasks
- Step 1: Plan the Technical Review Meeting
- Step 2: Review the Product
- Step 3: Conduct the Technical Review Meeting
- Step 4: Resolve Defects
- Step 5: Conduct Defect Causal Analysis
- Step 6: Resolve Open Issues
- Step 7: Document the Technical Review
- Step 8: Rework the Product
- Step 9: Review & Audit
- Step 10: Follow-up

Output
- Completed Peer Review Announcement and Report Form
- Reworked Product

Exit Criteria Met
- Issues Addressed
- Product Rework Completed
- Peer Review Announcement and Report Form completed

Figure 2-3. Technical Review Procedure for Work Products

2.5.2.5 Tasks. The tasks necessary to complete the technical review are described in the steps that follow:

Step 1. Plan the Technical Review Meeting

The review leader performs the following activities:

a. Identifies the review team and obtains appropriate management support, if needed.

b. Schedules and announces the meeting date, time, and place.

c. Distributes the Peer Review Announcement and Report (Form 1), the product to be reviewed and all necessary material to the reviewers. The review leader allows enough time for the reviewers to review the material and prepare for the meeting.

d. Determines if an overview of the product is needed. If an overview is needed, the overview can occur as part of the review meeting or as a separate meeting.

Step 2. Review the Product

Reviewers independently review the product and related materials in preparation for the technical review meeting. Reviewers may log defects on a Peer Review Defect Log (Form 2) and forward them to the recorder prior to the review meeting.

Step 3. Conduct the Technical Review Meeting

During the review meeting, the entire team reviews the software product, evaluating it relative to applicable guidelines, specifications and standards, and/or evaluating alternative problem solutions. Specifically, the review team performs the following tasks:

a. Examines the product under review and verifies that it complies with the specifications and standards to which it must adhere. May review all previously submitted Peer Review Defect Log (Form 2) during the meeting. All deviations from the specifications and standards are recorded. The Peer Review Defect Log (Form 2) in Appendix C may be used for this purpose.

b. Documents technical issues, related recommendations, and the individual(s) responsible for resolving issues.

c. Identifies other issues that must be addressed.

d. When deficiencies are sufficiently critical or numerous, recommend that the product undergo an additional review after rework.

Step 4. Resolve Defects

It is the responsibility of the review leader and author to resolve deficiencies. Open issues that cannot be resolved by the review leader and author are referred to the project lead.

Step 5. Conduct Defect Causal Analysis (Optional)

The review team reviews the product deficiencies to determine probable cause and to recommend steps to prevent similar defects from occurring in the future. If a defect was caused by an error in another baselined product, follow the appropriate configuration management process to correct that product.

Step 6. Resolve Open Issues

The review leader, author, and project lead resolve open issues.

Step 7. Document the Technical Review

Complete the Peer Review Announcement and Report (Form 1), including any causal analysis findings.

Step 8. Rework the Product

The author reworks the product.

Step 9. Review and Audit

SQA audits implementation of the Technical Review Procedure and all artifacts resulting from the Technical Review.

Step 10. Follow-up

Review Leader tracks Issues and Action Items to closure.

2.5.2.6 Output. The outputs of the technical review are the completed Peer Review Announcement and Report (Form 1) and the reworked product.

2.5.2.7 Exit Criteria. The exit criteria for the technical review are listed below:

a. All issues identified during the technical review meeting have been addressed.

b. Product has been reworked as necessary.

c. Peer Review Announcement and Report (Form 1) completed and a copy forwarded to SEPO.

2.5.3 Formal Inspection Procedure

2.5.3.1 Procedure Goal. The objective of an FI is to detect and remove defects. This is the most rigorous of the peer reviews. An expert mode of the FI Procedure, reference f, is available on the SSC San Diego PAL at http://sepo.spawar.navy.mil/.

2.5.3.2 Special Responsibilities. This section describes specific responsibilities of participants in the FI Procedure. Multiple roles can be assigned to FI participants as shown in Table 2-2.

TABLE 2-2. MULTIPLE ROLE MATRIX FOR FORMAL INSPECTION PARTICIPANTS

Primary Role	*MUST* Also Be	May Also Be	May *NOT* Also Be
Author	---	---	Moderator or Presenter or Recorder or Inspector or Observer
Moderator	Inspector	Presenter	Author or Recorder or Observer
Presenter	Inspector	Moderator	Author or Recorder or Observer
Recorder	Inspector	---	Author or Moderator or Presenter or Observer
Inspector	----	Moderator or Presenter or Recorder	Author or Observer
Observer	----	----	Any other role on the FI team

a. Inspectors. The inspectors in an FI are responsible for following the guidance provided in the inspector's checklist. (See Appendix C). The items in the inspector's checklist can be tailored to individual project needs and the type of work product to be inspected.

b. Moderator. The moderator is a key player in the FI Procedure and is responsible for conducting the inspection meeting. The role of the moderator requires skills in leadership, management and

team building, as well as technical competency in subject of the work product to be inspected. The moderator participates in all phases of the inspection process and has the following responsibilities:

1. Leads, manages, and coordinates the FI by providing guidance to inspection participants and demonstrating a positive attitude. The moderator also identifies team members and assigns roles to match and develop team member skills.
2. Follows guidance outlined in the moderator's checklist. (See Appendix C). The items in the moderator's checklist can be tailored to individual project needs and the type of work product to be inspected.
3. Facilitates the inspection meeting by effective prior planning including starting the meeting on time, focusing on the identification and classification of defects and open issues within the two-hour inspection period. The moderator may also facilitate the FI meeting by performing the following functions:
   a) Encouraging participation of all team members
   b) Following the rules and procedures of the FI Procedure
   c) Avoiding domination of the team
   d) Avoiding lengthy dialog with team members and discouraging "solution discussing"
   e) Promoting synergism of the team as a whole
   f) Limiting discussion of an issue to approximately two minutes
   g) Securing a "interrupt lock-out" meeting place
   h) Serving as an "Inspector" without special privileges
   i) Allowing team to decide defect classification by consensus

c. Author. The author prepares the work product to be inspected and prepares to tolerate and accept constructive criticism provided by inspection team members. The author participates in all phases of inspection and follows the guidance outlined in the author's checklist (See Appendix C). The items in this checklist may be tailored to individual project needs and the type of work product to be inspected.

d. Presenter. A primary responsibility of the presenter is to focus the attention of the inspection participants on details of the work product by paraphrasing and/or summarizing section(s) as appropriate. Note also the presenter participates in all functions required of other inspectors and follows the guidance outlined in the presenter's checklist (See Appendix C). The items in this checklist may be tailored to individual project needs and the type of work product to be inspected.

e. Recorder. The recorder is responsible for accurate and complete documentation of (1) defects reported by inspection team members, (2) defect classification, (3) open issues, (4) assigned action items, and (5) inspection feedback/lessons learned. This role requires a person with good writing skills and the ability to handle multiple tasks concurrently (e.g., listening, writing, pacing, verifying). The recorder uses the Peer Review Defect Log (Form 2) to record and classify defects and open issue items. Note the recorder participates in all functions required of other inspectors during the FI meeting and follows the guidance outlined in the recorder's checklist (See Appendix C). The items in this checklist may be tailored to individual project needs and the type of work product to be inspected.

During the inspection meeting, the recorder should sit next to the moderator with his/her writing hand on the same side as the moderator. This allows the moderator to assist the recorder in documenting information accurately and completely, as well as pacing the flow of verbal information to the writing speed of the recorder. Use of computer-based word processing system is suggested to facilitate rapid recording of this information. If the Peer Review Defect Logs (Form 2) have been merged prior to the Inspection meeting, the recorder will verify the information as the meeting proceeds and capture any additional defects that are presented.

The recorder should read back each defect or issue and its classification as it is recorded, rather than waiting until a number of individual defects or issues have been recorded. This approach ensures timely capture and verification of group consensus on reported defects/open issues. Further, information in the recorder's log is critical because it is what the author uses to understand and address the defect/issue during the rework phase. Finally, it is the responsibility of the recorder to interrupt inspectors if they are speaking too rapidly, or moving on to a new defect or issue before the recorder has had an opportunity to document and read back the previous defect/issue to inspection participants.

f. Observer - The observer attends the FI meeting for purposes of observing and recording information about participants, their roles, and group interaction. The person assigned this optional role should be well-versed in the FI Procedure and interested in overall improvement of the FI Procedure based on feedback and lessons learned by the participants. Appendix C includes a template the observer may use to record observations and a checklist for guidance.

2.5.3.3 Entry Criteria. The entry criteria are listed below:

a. The work product is ready for inspection

b. Plans, standards, guidelines, and checklists are available

c. Any focus areas for the particular product have been defined.

2.5.3.4 Input. The input to the FI is the product to be formally inspected.

2.5.3.5 Tasks. The tasks necessary to complete an FI are described in the steps that follow. Figure 2-4 presents a diagram of the FI Procedure steps.

Step 1. FI Planning Phase

The planning phase objectives are to (1) confirm the work product is ready for inspection, and (2) perform planning activities, which will result in a successful FI. Figure 2-5 presents a flow chart of the Planning Phase activities, which are described in more detail below:

a. Planning Phase Participants. Required participants for the planning phase include project management, moderator and author.

b. Planning Phase Activities. The following activities are performed

1. Assign proposed participants to specific roles

Project management selects the moderator and assists in the selection of other participants. Table 2-3 provides some suggestions for participant selection based on the type of product to be inspected. Once selected, the moderator assigns roles to participants since he/she is responsible for the successful preparation and conduct of the inspection. Individual expertise, skills, and personalities should be taken into account when assigning roles. The moderator meets with the presenter, recorder, and observer to discuss how they can each perform their assigned roles.

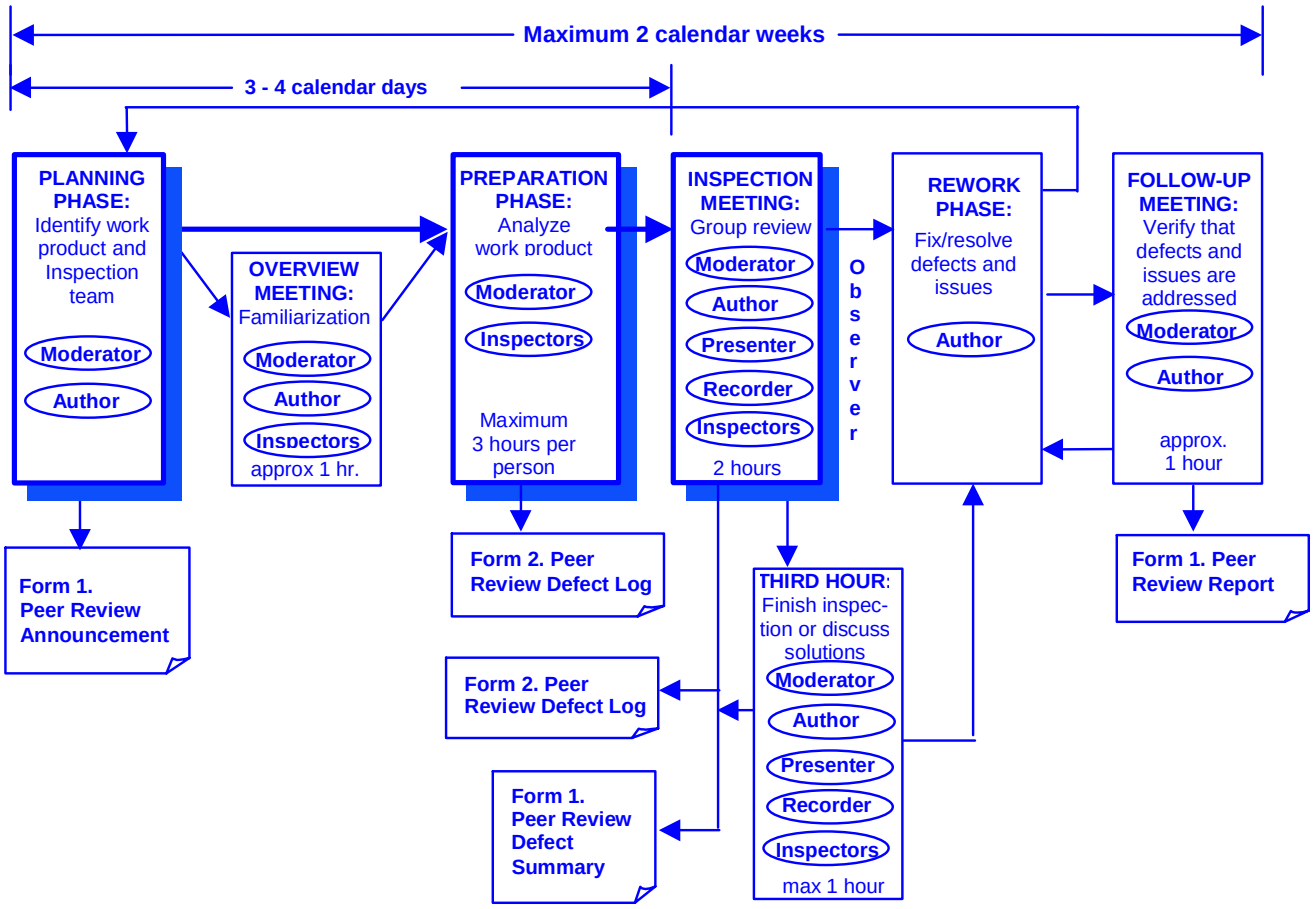


Figure 2-4. Formal Inspection Procedure Diagram

2. Determine the size of the work product to be inspected

   The moderator and author should reach a consensus on the amount of work product to be inspected during each two-hour inspection period. Table 2-4 provides guidelines for the volume of material participants can normally review for various types of work product. These guidelines are based on results of actual inspections and provide a basis for planning the appropriate number of two-hour inspection periods to be performed on a work product. Note, however, inspection rates may vary as a function of work product complexity and the experience of team members.

3. Establish requirement for overview meeting

   The moderator should consult with the author to determine the need for an overview meeting. The purpose of the overview is to educate members of the inspection team on the work product to be inspected, and to review the inspection process applicable to the work product and inspection team. If the moderator determines an overview is unnecessary, or has already been given during a previous FI of the work product, no meeting is scheduled.

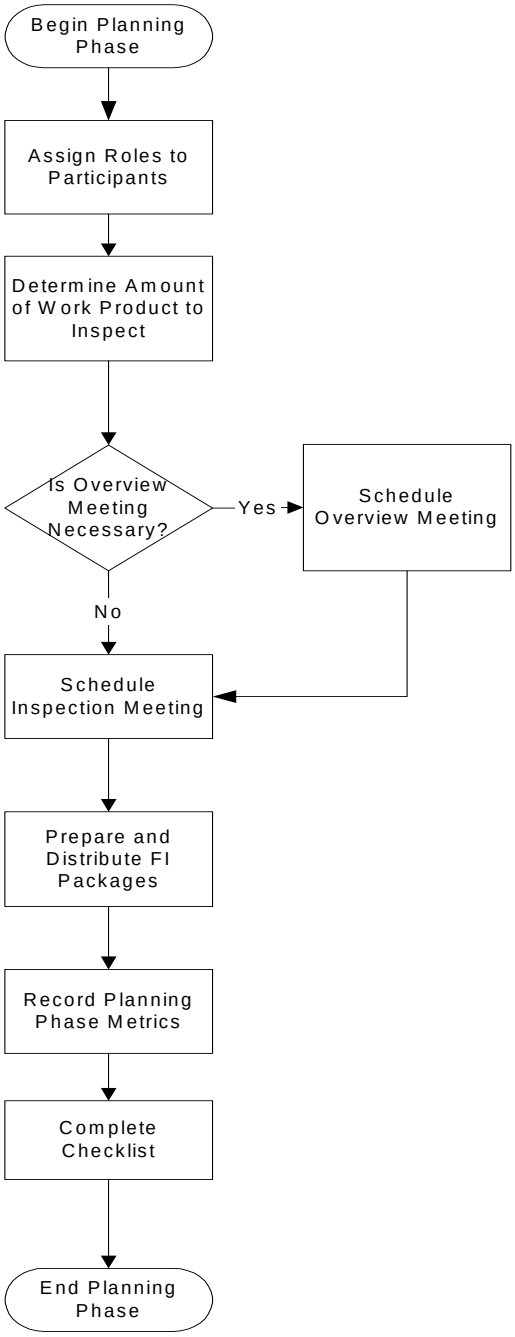


Figure 2-5. Flow Chart For FI Planning Phase

TABLE 2-3. SUGGESTED INSPECTION PARTICIPANTS BY TYPE OF WORK PRODUCT

PRODUCT TYPE	PRIMARY PARTICIPANTS	OTHER PATRICIPANTS (OPTIONAL)
System Requirements	Moderator, Systems Engineer (Author), Peer Systems Engineer, Software (S/W) Engineer (Designer), Test Engineer, User	SQA, Performance, Hardware, Operations, Algorithm Developers
Software Requirements	Moderator, Systems Engineer (Author), Peer Systems Engineer, S/W Engineer (Designer), Test Engineer, User	SQA, Performance, Hardware, Operations, Algorithm Developers
Preliminary Design	Moderator, Software Requirements (Author), S/W Engineer (Designer), Peer S/W Engineer, Test Engineer	Performance, Algorithm Developers, Operations, SQA, Maintenance, User
Detailed Design	Moderator, Preliminary Design (Author), S/W Engineer (Designer), Peer S/W Engineer, Test Engineer	Algorithm Developer, SQA
Source Code	Moderator, S/W Engineer (Author), Peer S/W Engineer, Detailed Designer, Test Engineer	Maintenance, Functional Designer, SQA
Test Plans	Moderator, Test Engineer (Author), Peer Test Engineer, Preliminary Design Author, S/W Engineer	SQA
Test Cases/Procedures	Moderator, Test Engineer (Author), Peer Test Engineer	SQA
User Documentation (Preliminary and Final)	Moderator, Author, S/W Engineer(s), Users	SQA

TABLE 2-4. RATE GUIDELINES FOR VARIOUS TYPES OF WORK PRODUCTS

Inspection Type	Preparation Rate	Inspection Rate
System Requirements (Specifications)	6 – 10 Pages Per Hour (PPH)	6 – 10 PPH
System Design (System/Segment Design)	6 – 10 PPH	6 – 10 PPH
Preliminary Design	10 – 20 PPH	10 – 20 PPH
Detailed Design	125 – 200 Lines of Program Design Language (PDL) Per Hr	125- 200 Lines of PDL Per Hr

Inspection Type	Preparation Rate	Inspection Rate
Source Code	125 – 200 Lines of code Per Hour (LPH)	125 – 200 LPH
Test Plan	10 – 20 PPH	10 – 20 PPH
Test Case/Procedure	10 – 20 PPH	10 – 20 PPH
User Documentation	10 – 20 PPH	10 – 20 PPH

4. Schedule overview and/or inspection meeting

   The moderator contacts designated participants to schedule a mutually acceptable date, time and location for the inspection meeting, and an overview meeting, if one is to be held. The overview meeting should be held within three working days of receipt of the initial (or corrected) work product.

5. Prepare and distribute FI packages to participants

   The moderator assembles the FI package that is a collection of forms and other documents given to each participant for their review and use in preparing for the FI. The package includes the work product to be inspected, relevant checklists, and any other documentation about the work product or standards relating to the work product, and the defect logs for recording defects and open issues. Work products being prepared for inspection should be line numbered, if appropriate, to facilitate logging defect locations. The following is a hypothetical example of an FI Package for the Navigation Sensor System Interface (NAVSSI) Project:

   a) Peer Review Announcement and Report (Form 1) with the Announcement portion completed (See Appendix C)

   b) Software Requirements Specification (SRS) for NAVSSI, pages 1-45.

   c) NAVSSI Interface Requirements Specification (IRS)

   d) MIL-STD 498 DID for an SRS

   e) Focus area checklists tailored to NAVSSI applications for the following quality areas: Clarity, Completeness, Compliance, Consistency, etc. (See Appendix B)

   f) Role checklists for FI participants (See Appendix C)

   g) NAVSSI Requirements Management Process for reference

   h) Blank Peer Review Defect Logs (Form 2) (See Appendix C)

   i) Summarized expectations of the results for the FI of the SRS (i.e., input to the overview meeting)

   To allow for adequate preparation time, it is recommended the package be assembled and distributed to participants a minimum of three calendar days before the scheduled inspection meeting date. The Peer Review Announcement and Report (Form 1) in Appendix C, provides essential information for participants concerning their roles, attendance at meetings, work product to be inspected, and due dates for defect logs.

6. Record time spent in the Planning Phase for future reference

The participants should record all time spent in Planning Phase activities for future reference. This data is useful for establishing metrics concerning the efficiency of efforts expended, and may also reveal ways to streamline preparation activities for subsequent FIs.

7. Complete the Planning portion of the participants' checklists

   Checklists of activities for participant roles in each phase of the FI Procedure are provided in Appendix C of this document. Each participant should follow their respective checklists for the Planning Phase, and document completion of their assigned activities. Any missing activities should be completed prior to entering the next phase. A decision about whether to continue should be made by the moderator regarding any significant problems in completing all required planning activities because of the potential impact on subsequent phases of the inspection.

Step 2. Overview Phase (If Required)

The objectives of this optional phase are to (1) educate participants on the work product to be inspected, and (2) review the FI Procedure applicable to the work product and inspection team. Figure 2-6 presents a flow chart of the Overview Phase activities that are described in more detail below:

a. Overview Phase Participants. Participants include any member of the team requiring the above types of information.

b. Overview Phase Activities. The following activities are performed during the Overview Phase:

   1. Provide an overview briefing on the work product to be inspected

      The overview briefing should clarify the work product by summarizing its purpose and objectives; identifying the intended audience for the work product; when it is to be used; where it applies; and how the objectives are to be accomplished. It is recommended the author provide the overview briefing, explaining what he/she expects to gain from the FI.

   2. Review assigned roles and plans for conducting the inspection

      The overview meeting should confirm each participant's understanding of his/her assigned role(s) and plans for conducting the FI. Specifically, the moderator should discuss and clarify any focus areas assigned to participants and ensure all participants understand how to use applicable forms and checklists.

   3. Address any questions and concerns

      It is recommended the moderator address any questions or concerns participants may have regarding their assigned roles, required responsibilities, or time commitments to the inspection.

   4. Record Overview Phase metrics

   5. Complete Overview portion of the participant's checklists

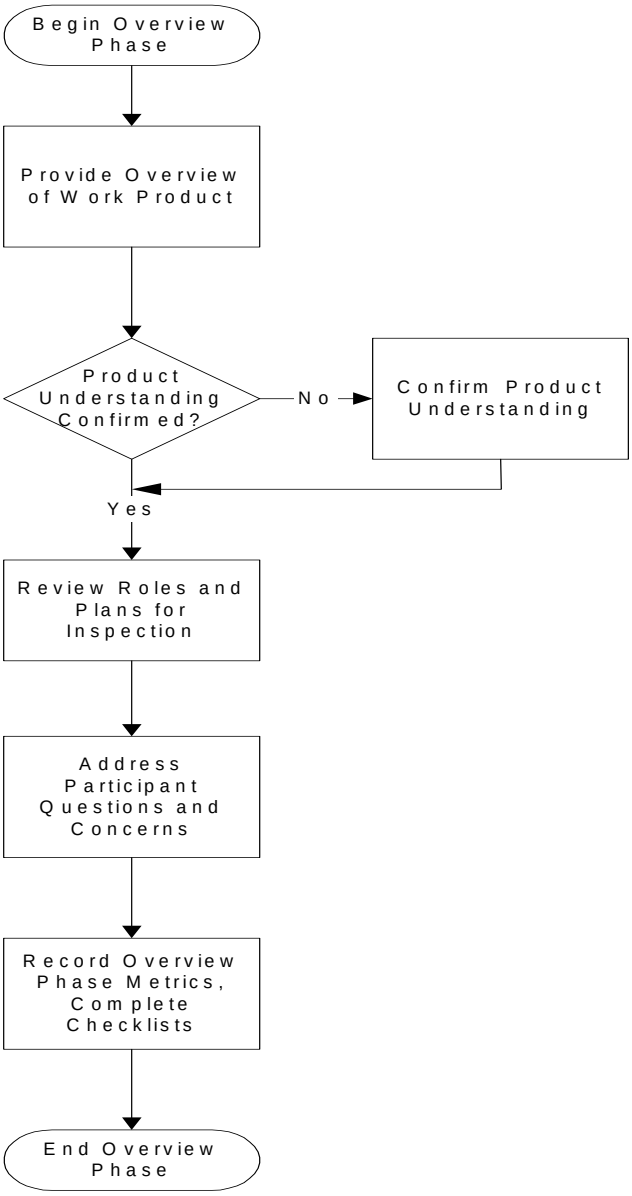


Figure 2-6. Flow Chart for Overview Phase

Step 3. Preparation Phase

The objectives of this phase are for inspectors to (1) examine the work product individually to detect defects and (2) classify each defect detected. Figure 2-7 presents a flow chart of the Preparation Phase activities that are described in more detail below:

a. Preparation Phase Participants. Participants include the moderator and all other inspectors.

b. Preparation Phase Activities. The following activities are performed during the Preparation Phase:

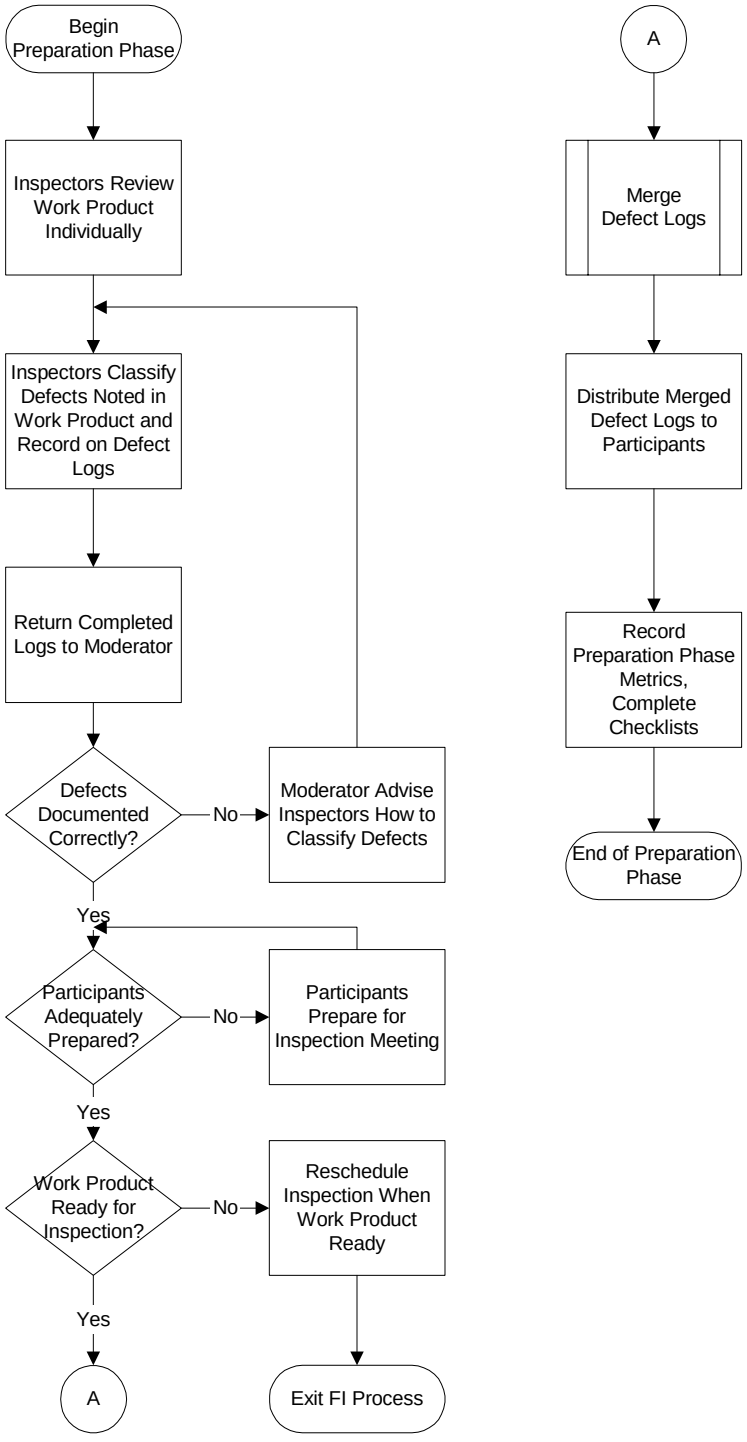


Figure 2-7. Flow Chart for Preparation Phase

1. Review the work product

   Inspectors review the work product concentrating on their assigned focus areas.

2. Classify and record all defects

   Inspectors classify and record defects as specified in the FI package and overview meeting. Individual inspectors complete the Peer Review Defect Log (Form 2) to document and record defects. Classify all defects identified during review of the work product as major or minor. Guidelines for recording defects are provided in Appendix C. Redline errors should be noted in a hard copy of the work product and submitted at inspection meeting.

3. Return completed defect logs by the due date

   Inspectors are expected to return their completed defect logs to the moderator by the due date.

4. Confirm defects have been documented correctly

   The moderator reviews individual defect logs submitted by inspectors to confirm defects have been recorded correctly (i.e., comply with prescribed standards). The moderator makes a decision whether to proceed in cases where significant errors exist either in the description or classification of defects submitted by the inspectors.

5. Ensure all participants are adequately prepared for the inspection

   The moderator, based on a review of the defect logs submitted, makes a decision whether to proceed. If two or more participants indicate they are not prepared for their role in the inspection, based on the first "Yes/No" check box on the first page of the defect log, the inspection meeting may be postponed until all participants are ready.

6. Confirm readiness of the work product for inspection

   The moderator determines if the product is ready for inspection based on the second "Yes/No" check box on the first page of the defect log. Too many defects may indicate the work product is not of adequate quality to merit having an inspection meeting. If this is determined to be the case, the FI is aborted until the work product is more mature.

7. Merge and distribute the Peer Review Defect Logs (Form 2)

   The FI moderator or recorder uses the FI Log Merge Procedure (reference c or d), available at http://sepo.spawar.navy.mil/ (open the SW Topics folder and go to the sample software forms sub-page), to merge the individual inspector's defect logs into a single consolidated defect log, and distributes the merged defect log to inspection participants.

8. Record Preparation Phase metrics

9. Complete the Preparation portion of the participant's checklists

Step 4. Inspection Phase

Objectives of the Inspection Phase are to (1) attain group consensus regarding reported defects and issues affecting the work product, (2) determine by group consensus if the work product requires re-inspection, and (3) discuss and document lessons learned from the process by obtaining feedback from inspection participants and observer, as applicable. Section 2.5.3.2 provides detailed guidance for each FI role and

how best to participate as an inspector during this phase. Figure 2-8 presents a flow chart of the Inspection Phase activities, which are described in more detail below:

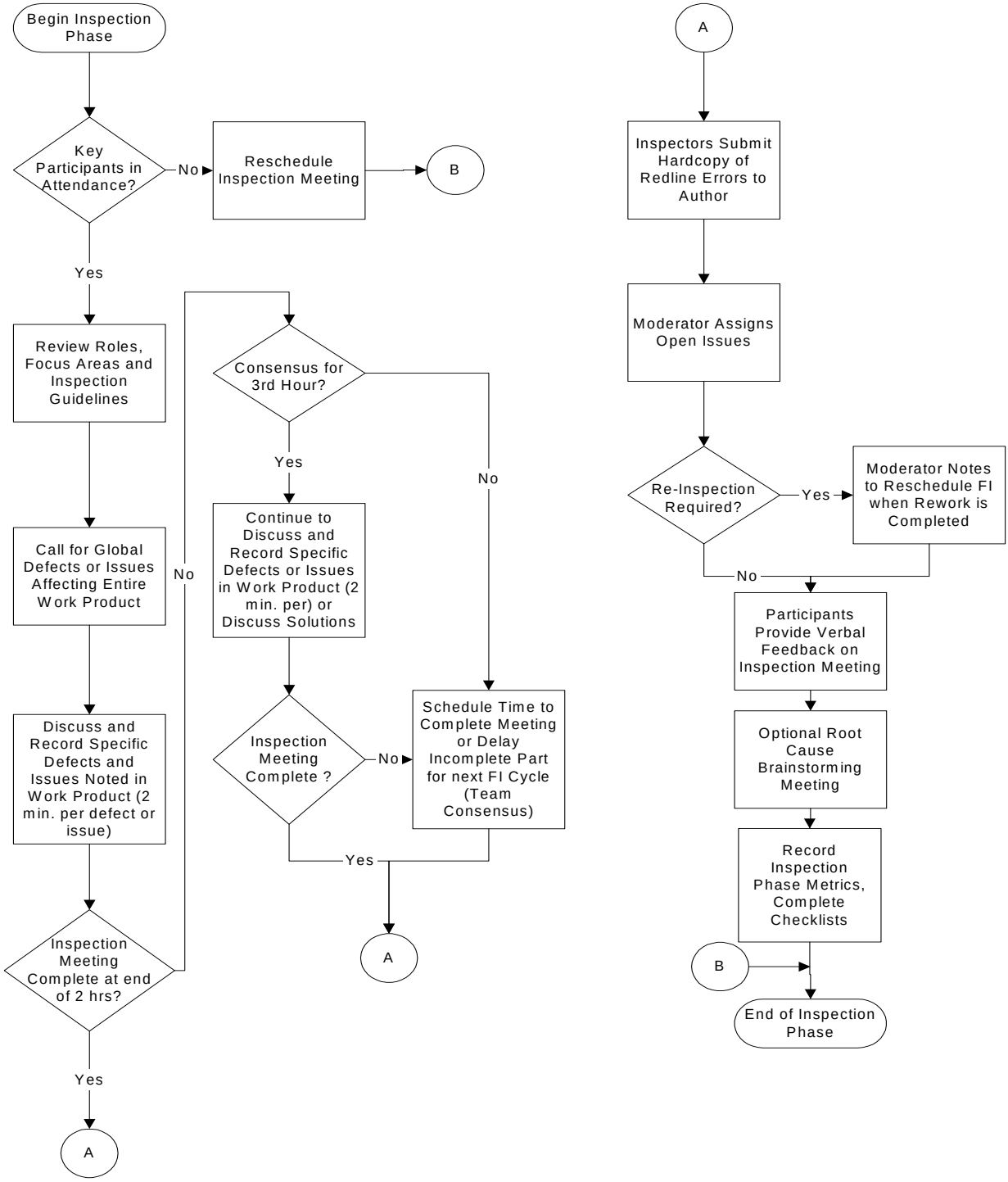


Figure 2-8. Flow Chart for Inspection Phase

a. Inspection Phase Participants. Required participants for this phase include the moderator, author, recorder, and inspectors. An observer (optional) may also serve on the inspection team.

b. Inspection Phase Activities. The following activities are performed as part of the inspection meeting:

   1. Determine if there are enough participants to conduct the inspection

      At the scheduled time for the inspection meeting, the moderator determines if there are enough participants and delays the inspection if there are three or fewer inspectors present.

   2. Review roles, focus areas and guidelines for conducting the inspection

      The moderator reviews assigned roles, focus areas, and guidelines for conducting the inspection with all participants.

   3. Call for global defects or issues affecting the entire work product

      The moderator ensures the presenter begins the inspection meeting by asking participants to present and discuss global defects or issues, which affect the entire work product. The term "global" refers to defects or issues which permeate the work product.

   4. Call for specific defects and issues to be reviewed and recorded

      The moderator or presenter ensures all participants are given an opportunity to report individual defects and issues to be recorded. This is accomplished by a "round-robin" technique in which the presenter initially calls upon a different inspector to report individual defects or issues listed on his/her defect log as each new section or portion of the work product is being inspected. Other inspectors are then called upon in turn to add to what the first inspector for that section addressed, if required. The group classifies each defect as major or minor. If unable to make this determination, the defect is classified as open. Note: a maximum of two minutes is allowed to discuss each defect or open issue. No discussion of solutions to issues or defects is allowed at this time. At the end of

   5. Determine if additional time (a third hour) is required

      The moderator should ask inspection participants if they are willing to remain for an additional period of time (called the third hour) if the work product cannot be adequately reviewed during the two-hour period allocated for the FI. If participants indicate they cannot remain for the third hour, the moderator may schedule another meeting to complete the inspection of the work product, or delay the incomplete part until the next FI cycle. In either case, the moderator should obtain group consensus prior to choosing one of the above alternatives.

   6. Use third hour to complete remaining inspection activities

      If participants agree to extend the inspection to a third hour, the moderator should concentrate his/her efforts on such activities as: completing inspection of the work product, classifying and recording defects, resolving issues, and assigning action items. If the team chooses to do so, the third hour may also be used to discuss solutions.

   7. Submit "redline" errors to author for corrective action

At the conclusion of the inspection, each participant submits a copy of the work product to the author with redline errors (e.g., spelling, punctuation, grammar, syntax) noted for corrective action. Redline errors are not considered defects or open issues, and as such, are not recorded on defect logs.

8. Assign open issues

   The moderator assigns all open issues to appropriate participants at the conclusion of the meeting. The moderator follows-up with assigned persons one or two working days after the inspection meeting to establish realistic timeframes for open issue(s) that cannot be readily resolved.

9. Decide if the work product should be re-inspected

   At the conclusion of the meeting, the moderator should obtain group consensus regarding the need to re-inspect the work product. If the group is unable to arrive at consensus, the moderator makes the decision before proceeding to next phase.

10. Solicit feedback from participants regarding the inspection

    At the conclusion of the meeting, the moderator should obtain feedback from inspection participants and the observer (if assigned) to generate lessons learned and recommended improvements to the FI Procedure. Although use of an observer is optional, inexperienced inspectors may benefit from the impartial observations/comments provided by the observer concerning their effectiveness during the inspection meeting.

11. Conduct optional root cause brainstorming meeting

    The purpose of the brainstorming meeting is to review the major defects and brainstorm possible/probable causes and how to prevent them from reoccurring. Discussion on each defect should be limited to approximately three minutes. The recorder should record the results of this activity for inclusion with the FI final report. As process maturity increases, these results will be used to systematically improve processes to prevent defects from being injected into the work products.

12. Record Inspection Phase metrics

13. Complete Inspection portion of the participant's checklists

Step 5. Rework Phase

The objectives of this phase are to (1) correct all defects, and (2) resolve all open issues. Figure 2-9 presents a flow chart of the Rework Phase activities, which are described in more detail below:

a. Rework Phase Participants. Participants for the Rework Phase include the author (required) and project management/others (optional).

b. Rework Phase Activities. The following activities are performed in this phase:

   1. Estimate due dates for resolution of open issues

      Once an open issue has been assigned, the responsible team member should provide feedback to the moderator concerning its due date for resolution. In some cases, additional time beyond the suggested two-calendar week period for conducting FIs may be required if responsibility for resolving open issues is beyond the assigned person's authority. Similarly, the two-calendar week period may not be realistic in situations where activities outside of SSC San Diego are responsible for correcting defects or resolving open issues.

2. Correct all defects identified in the work product

   Within reasonable limits, the author should correct all major and minor work product defects identified during the FI. In some cases, a work product containing minor defects is acceptable as determined by project needs; for example, cost and/or schedule

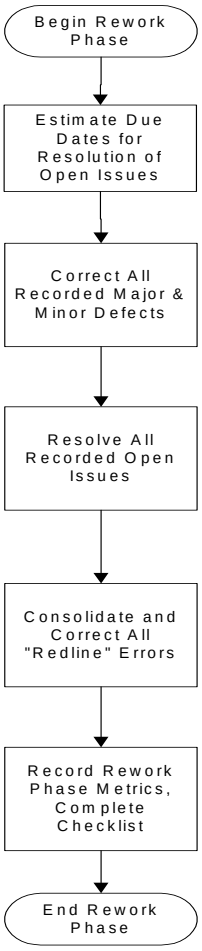


Figure 2-9. Flow Chart for Rework Phase

constraints imposed by the sponsor or customer. If such constraints apply, the author should attempt to correct all major defects first, followed by minor defects.

3. Resolve all open issues for the work product

   Team members assigned responsibility for open issues should make every effort to resolve them in accordance with the due dates provided to the moderator. Note, however, some open issues may not be resolved within the suggested two-calendar week period for completing the FI Procedure.

4. Correct all "redline" errors identified in the work product

   The author should correct all "redline" errors submitted by individual inspectors in the work product.

5. Record Rework Phase metrics

6. Complete Rework portion of the participant's checklists

Step 6. Follow-Up Phase

The objectives of this phase are to (1) verify all defects planned for rework have been corrected, (2) confirm all open issues have been resolved, (3) ensure all "redline" errors have been corrected, and (4) close-out the inspection. Figure 2-10 presents a flow chart of the Follow-Up Phase, which is described in more detail below:

a. Follow-up Phase Participants. Required participants for this phase include the author and moderator.

b. Follow-up Phase Activities. The following activities are performed during this phase:

   1. Schedule the follow-up meeting with the author of the work product

      The moderator makes a decision regarding the need for a follow-up meeting with the author of the work product. If the moderator believes the quantity or quality of corrections is unsatisfactory, the moderator returns the work product to the author for further rework.

   2. Verify all defects planned for rework have been corrected

      The moderator reviews the defect log with the author to verify all defects planned for correction have been corrected. In cases where a corrective action is questionable, the moderator will consult with project management, inspection team members or co-workers as required. Unsatisfactory corrections require return to the Rework Phase.

   3. Verify all open issues planned for resolution have been resolved

      The moderator reviews the defect log with the author to confirm all open issues planned for resolution have been resolved, consulting with project management, as appropriate. Unsatisfactory resolution of open issues requires return to the Rework Phase.

   4. Verify all "redline" errors have been corrected during rework

      The moderator reviews the reworked product to verify correction of redline errors. Support from an administrative assistant is recommended when verifying the correction of "redline" errors.

   5. Log remaining open issues or defects into project tracking system

      Any defects or open issues not resolved need to be transitioned into whatever process is used to track action items (e.g., Work Breakdown Structure) before the FI can be logged as closed. This should be a rare occurrence.

   6. Record Follow-Up Phase metrics

   7. Complete the Follow-Up portion of the participant's checklists

      Once all of the participant's checklists are completed, the moderator collects them and transcribes the hours logged onto the Peer Review Announcement and Report (Form 1).

   8. Complete and distribute the FI documentation

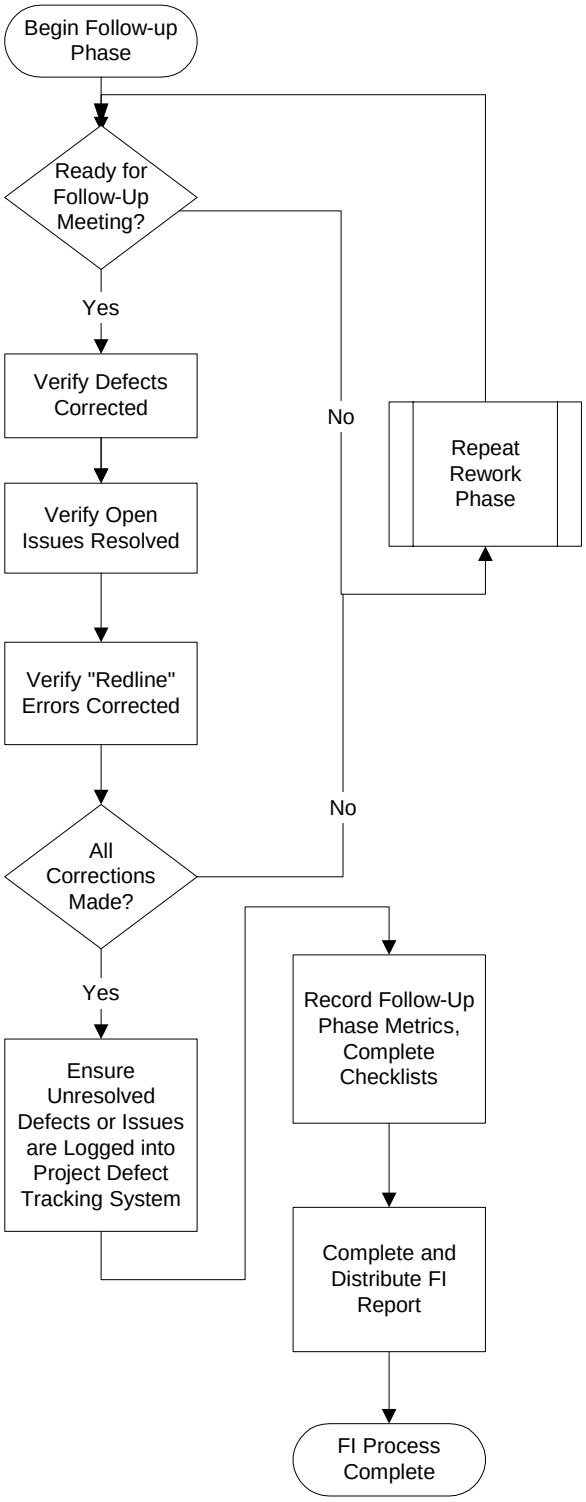


Figure 2-10. Flow Chart for Follow-up Phase

The moderator is responsible for completing the Report portion of the Peer Review Announcement and Report (Form 1) to close-out the inspection. This form provides the following statistical data: (1) total time spent by participants, (2) number of defects detected/corrected, and (3) number of open issues detected/resolved. Form 1 is also used to record the date for completing the inspection and re-inspection of the work product, if required. The consolidated defect log, participant checklists, root cause brainstorming results, if conducted, and lessons learned should be included as attachments to the Peer Review Announcement and Report (Form 1). The package is distributed to inspection participants, project management, and SEPO. The FI cannot be considered closed out until the report is completed and distributed.

2.5.3.6 Output. The output of the FI is the completed Peer Review Announcement and Report (Form 1) with attachments and the completed reworked product.

2.5.3.7 Exit Criteria. The following criteria apply when confirming completion of the FI Procedure:

a. The work product has been inspected and all rework is completed

b. All metrics have been collected

c. The FI documentation has been completed and distributed

d. Improvement suggestions have been forwarded to SEPO.

2.6 MEASURE PEER REVIEW PROCESS

Measurements are made and used to determine the status of the peer review activities. Examples of measurements are listed below:

a. Number of peer reviews performed compared to the project's SDP

b. Overall effort expended on peer reviews compare to the plan (staff hours)

c. Number of products reviewed compared to the project's SDP.

Data that should be collected and used to measure the effectiveness of the Peer Review Process are listed below:

a. Identification of the product reviewed

b. Size of the product

c. Size and composition of the review team

d. Length of the review meeting

e. Types and number of defects found and fixed

f. Results of any defect causal analysis done as part of review

g. Hours spent in rework effort

h. Hours spent in the overall review effort

i. Suggestions for improving the process.

Appendix C of this document provides a Peer Review Announcement and Report (Form 1) which includes most of the items listed above and shall be used to facilitate collection of required data resulting from conducting walkthroughs, technical reviews, and formal inspections.

2.7 REVIEW AND AUDIT PEER REVIEW PROCESS

To ensure that peer reviews are being properly conducted, reviews or audits of peer reviews are required. The purpose of a review or audit of the Peer Review Process is to verify that peer reviews are conducted as planned. The audit or review should be planned for and scheduled in the project's SDP or SQA Plan. It is an SQA responsibility to perform the review or audit. As part of the SQA review or audit, verification is made that project personnel are trained for their roles in participating in the peer reviews. Peer Review Process audits will reveal if the peer review participants prepare for the peer reviews, conduct the peer reviews in accordance with the process, and perform the required follow-up actions. An audit will also reveal if peer review data collected and reported is complete, accurate, and timely.

This page intentionally left blank.

SECTION 3. PROCESS SUPPORT

This section describes assistance and training provided by SEPO, along with automated tools and techniques that may be used to successfully implement peer reviews of software engineering-related work products at SSC San Diego. Users and prospective users of this process are encouraged to contact SEPO for further information on available training, consultant services and automated process support tools.

3.1 ASSISTANCE

SEPO supports users and prospective users of the Peer Review Process by providing the following assistance:

a. Training personnel in the Peer Review Process

b. Implementing the process for on-going or planned efforts

c. Tailoring the process to meet the needs of a particular project

d. Identifying projects that have already implemented this process.

3.2 TRAINING

SEPO provides a Peer Review Workshop for technical and support code personnel at SSC San Diego. During this workshop, students learn the Peer Review Process in detail through formal instruction and by participating in two FI exercises. Students gain experience and understanding of assigned roles, group interaction, and guidance for effectively planning and conducting FIs. These workshops are offered on a regularly scheduled basis at SSC San Diego. All participants should receive peer review training prior to implementing this process on their projects. Further information is available on the SSC San Diego PAL.

3.3 TOOLS

A variety of tools are available to assist users in streamlining and tailoring the Peer Review Process to best meet individual project needs. General guidance for conducting FIs appears in Table 3-1. Guidelines concerning the use of metrics appear in Table 3-2. The measurement guidelines are based on experience and lessons learned from organizations outside SSC San Diego, and are intended for use when initially implementing the FI Procedure. After further experience, users may modify these guidelines based on their own lessons learned and individual project needs

Tools available to assist with the Peer Review Process provide a means for streamlining various phases of the Walkthrough and Technical Review Procedures and the FI Procedure as well as reducing the time spent by participants in documentation related tasks. The following tools are suggested as aides:

a. Word processing software may be used to perform the following functions:

   1. Tailoring data collection forms and checklists to meet individual project needs

   2. Providing direct inputs into forms used in the Preparation and Inspection Phases

   3. Editing inputs provided by individual inspectors

   4. Merging inputs from multiple inspectors—resulting in a consolidated defect log for use during the inspection meeting. (See the FI Log Merge Procedure available at URL

TABLE 3-1. ADDITIONAL GUIDELINES FOR CONDUCTING FORMAL INSPECTIONS

GUIDELINE	PURPOSE/RATIONALE
1. Use tailored focus area checklists for different types of work products.	Formal inspections may be tailored to any work product by selecting an appropriate focus area checklist for the type of work product involved. (See Appendix C for example checklists.)
2. Use defined standards.	Inspection team members are advised to stick to the project's documented standards for a given work product or to update the standards to reflect how they are being used.
3. Avoid discussing solutions to reported defects and issues.	This guideline is suggested to economize time spent in the two-hour inspection period, thereby allowing efforts to focus on reporting and classifying defects, and identifying open issues.
4. Report and document all detected defects and open issues.	This guideline is recommended to avoid temptation of the author/other participants to establish an "off-line" list of defects or issues to be treated separately from those documented in the defect log.
5. All participants should avoid negative criticism and comments.	This guideline is cited because of potential problems when one or more participants engages in counter-productive behavior which disrupts the objective of detecting defects, and recognizing and resolving open issues related to the work product.
6. Adhere to a two-minute time limit for discussions.	This guideline is suggested to avoid prolonged discussions of a defect, its classification, or an open issue during the FI meeting.
7. Avoid assignment of inspection participants to multiple, incompatible roles.	This guideline is offered to avoid conflicts that may arise by assigning participants to multiple, incompatible roles (e.g., the same person serving as author and moderator, or moderator and recorder). Refer to Table 2-2 for additional guidance in this area.
8. Minimize last minute changes to inspection team member roles.	This guideline emphasizes the need for stabilizing personnel assignments so all participants can prepare and effectively perform their assigned role during the inspection. With inexperienced users, last minute changes can create a major impact to the effectiveness of inspection team member performance.

http://sepo.spawar.navy.mil. Expand the SW Topics folder in the left frame and click on the sample software forms link. Scroll down to FI forms.)

5. Editing the consolidated log during the inspection meeting as submitted defects and open issues are discussed and approved.

b. Database management software may be used to track defects and open issues to correction and resolution

c. Spreadsheet software may be used to compute and chart metrics when the inspection is completed.

TABLE 3-2. GUIDELINES FOR USE OF METRICS WITHIN FORMAL INSPECTIONS

GUIDELINE	PURPOSE/RATIONALE
1. Use data from defect logs to generate initial metrics. Data reported on defect logs should be used to generate an initial set of metrics concerning the FI Procedure. The following simple metrics are suggested for initial use by projects at SSC San Diego:	
a. Actual inspection rates for the type of work product inspected.	Use the defect logs to record the Pages Per Hour (PPH) or Lines of Code Per Hour (LPH) covered during preparation and inspection phases. This metric can be useful in estimating time requirements for future inspections involving similar work products.
b. Percentage of total defects submitted vs. those approved during inspection meeting.	Use the defect logs to obtain separate counts of the total defects approved by inspection participants (numerator) and the total number submitted (denominator), then compute the percentage. This metric can be useful in gauging performance relative to the "hit rate" for submitted defects.
c. Percentage of total open issues reported vs. those resolved following the inspection.	Use the defect logs to obtain separate counts of the total number of open issues resolved at the time the inspection is closed-out (numerator) and the total number detected during the inspection meeting (denominator), then compute the percentage. This metric provides an index of how many open issues were resolved within the prescribed (i.e., 2-week) time limit for completing an FI.
d. Average time spent per recorded defect and/or open issue.	Use the final inspection report to obtain the total number of defects and open issues recorded during the inspection, and divide it by the total hours spent in detecting, correcting and resolving all defects and open issues. This metric is useful for determining the time (cost) per defect found and resolved.
2. Compare metrics from your project to those of other SSC San Diego projects. Metrics from SSC San Diego projects that have implemented the FI Procedure are available at SEPO. This data, which is derived from FI reports submitted to SEPO, provide a basis for comparing metrics from your project to those of other projects—both individually and center-wide. Providing inspection results to SEPO is also important for maintaining the SSC San Diego Peer Review metrics database.	
3. Limit the use of metrics during initial inspection efforts; then expand. Projects using the FI Procedure for the first time should limit their use of metrics to those described in a-d above.	

d. Electronic media may be used to save significant time, particularly in cases where inspection participants are geographically dispersed. Email may be used to transfer forms and instructions to participants and return completed forms and reports to SEPO and other interested parties.

This page intentionally left blank.

APPENDIX A. PEER REVIEW KPA TRACEABILITY MATRIX

Goal/ Key Practice	Description	SSC San Diego Coverage (SEPO)
Goal 1	Peer review activities are planned.	Peer Review (PR) Process Section 2.3
Goal 2	Defects in the software work products are identified and removed.	PR Process Section 2.5
Commitment 1	The project follows a written organizational policy for performing peer reviews	PR Process Section 1.1, Software Engineering Process Policy (SEPP)
Ability 1	Adequate resources and funding are provided for performing peer reviews on each software work product to be reviewed.	PR Process Section 2.3, SDP Template
Ability 2	Peer review leaders receive required training in how to lead peer reviews.	PR Process Section 2.3, : PR Flyer
Ability 3	Reviewers who participate in peer reviews receive required training in the objectives, principles, and methods of peer reviews.	PR Process Section 2.3, : PR Flyer
Activity 1	Peer reviews are planned, and the plans are documented.	PR Process Section 2.3, SDP Template
Activity 2	Peer reviews are performed according to a documented procedure	PR Process Section 2.5, SDP Template
Activity 3	Data on the conduct and results of the peer reviews are recorded.	PR Process Section 2.6; Peer Review database
Measurement 1	Measurements are made and used to determine the status of the peer review activities.	PR Process Section 2.6
Verification 1	The software quality assurance group reviews and/or audits the activities and work products for peer reviews and reports the results.	PR Process Section 2.7, : SQA Process; SQA Plan Template

This page intentionally left blank.

APPENDIX B. SOFTWARE PRODUCT CHECKLISTS

The checklists in this section are for reproduction and distribution to review/inspection team members.

The checklists are provided to assist inspection participants in their review of plans, documents and source code work products typically produced during the software engineering life cycle. The checklists are intended for use in addition to guidance provided in software development and documentation standards. The subsections in the checklists may be assigned as focus areas to various review participants.

The checklists may be tailored to individual project needs and the type of work product to be inspected.

This page intentionally left blank.

TABLE B-1. SOFTWARE DEVELOPMENT PLAN CHECKLIST

Item	Yes/No	Comments
CLARITY		
1. Is the primary purpose of the Software Development Plan (SDP) and each section being met?		
2. Is the terminology consistent and understandable from the perspective of the program manager, sponsor, software project manager (you), and developers?		
3. Are resource allocations, schedules, and milestones clear and unambiguous?		
COMPLETENESS		
1. Are the project's organizational structure, resource allocations, schedules, and milestones complete?		
2. Are risks adequately addressed and solutions identified?		
3. Have results of tradeoff analysis been reported and justifications for decisions been provided?		
4. Have assumptions about intended sequences, resources, and milestones been stated?		
5. Are appropriate processes identified and adequately explained with procedures for tracking their use and application? Processes are: (a) cost, size, and schedule estimation; (b) formal inspections; (c) metrics; (d) requirements definition; (e) design; (f) testing; (g) SQA; (h) CM; (i) IV&V; (j) project planning; (k) risk management; (l) project reviews; (m) software capability evaluation, (n) integration; and (o) training.		
LEVEL OF DETAIL		
1. Is the information provided of necessary and sufficient detail for the intended audience?		
2. Have all "TBSs" and "TBDs" been resolved?		

Item	Yes/No	Comments
3. Is there superfluous information or information that doesn't belong in this part of document?		
CORRECTNESS		
1. Do resource allocations, schedules and milestones track with estimates/results for similar efforts?		
2. Is there substantiating information to support proposed allocations, schedules, and milestones?		
3. Are schedules/milestones displayed to show absence of bottlenecks, conflicts, and unnecessary gaps?		
MAINTAINABILITY		
1. Is material presented to support the SDP to be a living, dynamic document that can be maintained?		

TABLE B-2. SYSTEMS REQUIREMENTS CHECKLIST

Item	Yes/No	Comments
CLARITY		
1. Are goals of the system defined?		
2. Are requirements specified in an implementation-free way that does not obscure the original requirements?		
3. Are implementation method and technique requirements kept separate from functional requirements?		
4. Is the terminology consistent with the user and/or sponsor's terminology?		
5. Are requirements clear and unambiguous or are there aspects of requirements that you do not understand?		
COMPLETENESS		
1. Are requirements stated completely—addressing relevant aspects, yet tolerant of temporary incompleteness?		
2. Has a feasibility analysis been performed and documented?		
3. Is the impact of not achieving the requirements documented?		
4. Have trade studies been performed and documented?		
5. Have the security issues of hardware, software, operations personnel and procedures been addressed?		
6. Has the impact of the project on users, other systems, and the environment been assessed?		
7. Are required functions, external interfaces, and performance specifications prioritized by the needed date?		
COMPLIANCE		

Item	Yes/No	Comments
1. Does this document follow project's system documentation standards? DoD standards?		
CONSISTENCY		
1. Are requirements stated consistently without contradicting themselves/other system's requirements?		
FUNCTIONALITY		
1. Are all functions clearly and unambiguously described and alphabetized?		
2. Are all described functions necessary and sufficient to meet mission/system objectives?		
INTERFACES		
1. Are all external interfaces clearly defined?		
2. Are all internal interfaces clearly defined?		
3. Are all interfaces necessary, sufficient, and consistent with each other?		
MAINTAINABILITY		
1. Have the requirements for system maintainability been specified?		
2. Are requirements written to be as weakly coupled as possible?		
PERFORMANCE		
1. Are all required performance specifications and margins listed?		
RELIABILITY		
1. Are there reliability requirements?		
2. Are there error detection, reporting, and recovery requirements?		
3. Are undesired events considered and their required responses specified?		

Item	Yes/No	Comments
4. Have assumptions about the intended sequence of functions been stated?		
5. Do these requirements adequately address survivability of the system?		
TESTABILITY		
1. Can the system be tested, demonstrated, inspected, or analyzed to show that it satisfies requirements?		
2. Are requirements stated precisely to facilitate specification of system test success criteria and requirements?		
TRACEABILITY		
1. Are all functions, structures, and constraints traced to mission/system objectives?		
2. Can all requirements be allocated to hardware, software, and operations personnel, and procedures?		

TABLE B-3. SOFTWARE REQUIREMENTS CHECKLIST

Item	Yes/No	Comments
CLARITY		
1. Is the terminology consistent with the users' and/or sponsors' terminology?		
2. Are software requirements clear and unambiguous?		
3. Have requirements been stated in terms of inputs, outputs, and processing for each function?		
COMPLETENESS		
1. Are required attributes, assumptions, and constraints of the program set completely listed?		
2. Have all requirements and constraints been assigned a priority?		
3. Have criteria for assigning requirement priority levels been defined?		
COMPLIANCE		
1. Does documentation follow project and/or DoD standards?		
CONSISTENCY		
1. Are requirements consistent with each other and with other requirements in related documents?		
2. Are requirements consistent with the actual operating environment and system specification?		
DATA USAGE		
1. Have critical values for all internal data items been specified? (e.g., data type, rate, units, accuracy, resolution, limits, range)		
2. Have data objects and their component parts been specified?		

Item	Yes/No	Comments
FUNCTIONALITY		
1. Are all described functions necessary and sufficient to meet the mission/system objectives?		
2. Are all inputs/processing/outputs to a function necessary and sufficient to perform requirements?		
INTERFACE		
1. Are inputs and outputs for all interfaces necessary and sufficient?		
2. Are interface requirements between hardware, software, personnel, and procedures included?		
LEVEL OF DETAIL		
1. Have functional requirements been described in sufficient detail for design work to begin?		
2. Have performance requirements been described in sufficient detail for design work to begin?		
MAINTAINABILITY		
1. Are requirements weakly coupled (i.e., changing a function will not have adverse/unexpected effects throughout system)?		
2. Have system specification maintainability requirements been levied to software functional requirements?		
PERFORMANCE		
1. Have resource and performance margin requirements been allocated to each function?		
RELIABILITY		
1. Have quality factors been specified as measurable requirements or		

Item	Yes/No	Comments
prioritized design goals?		
2. Are undesired events considered and their required responses specified?		
TESTABILITY		
1. Can the program set be tested, demonstrated, analyzed, or inspected to show that it satisfies requirements?		
2. Have test methods (test, demonstration, analysis, or inspection) been stated for each software requirement?		
TRACEABILITY		
1. Are all functions, structures, and constraints traced to requirements, and vice versa?		

TABLE B-4. SOFTWARE PRELIMINARY DESIGN CHECKLIST

Item	Yes/No	Comments
CLARITY		
1. Is architecture, including data flows, control flows and interfaces, clearly representative of design?		
2. Are all of goals, assumptions, constraints, decisions, and dependencies for this design documented?		
COMPLETENESS		
1. Have design trade-offs been identified with criteria for design selected vs. trade-offs documented?		
2. Has design modeling been performed and documented?		
COMPLIANCE		
1. Does documentation follow project and/or DoD standards?		
CONSISTENCY		
1. Are data elements, procedures, and functions named and used consistently throughout program set?		
2. Does design reflect the actual operating environment, hardware, and software?		
CORRECTNESS		
1. Is design feasible from schedule, budget, and technology standpoints?		
DATA USAGE		
1. Is the conceptual view for all composite data elements, parameters, and objects documented?		

Item	Yes/No	Comments
2. Has management and use of shared and stored data been clearly described?		
FUNCTIONALITY		
1. Does the specification for each module fully implement functionality documented for software requirements?		
INTERFACES		
1. Is the operator interface designed with the user in mind (i.e. vocabulary, useful messages, Sec. 508)?		
2. Have number and complexity of interfaces been minimized?		
LEVEL OF DETAIL		
1. Are all possible states or cases considered?		
2. Is design expressed in sufficient detail to proceed to detailed design?		
MAINTAINABILITY		
1. Is the design modular?		
2. Do modules have high cohesion and low coupling?		
PERFORMANCE		
1. Has performance modeling been performed and documented when appropriate?		
2. Have critical path(s) of execution been identified and analyzed?		
RELIABILITY		
1. Does design provide for error detection and recovery (e.g. input checking)?		

Item	Yes/No	Comments
2. Are all error conditions specified completely and accurately?		
TESTABILITY		
1. Can the program set be tested, demonstrated, analyzed, or inspected to show that it satisfies requirements?		
2. Can the program set be integrated and tested in an incremental manner?		
TRACEABILITY		
1. Are all parts of the design traced back to requirements document?		
2. Can all design decisions be traced back to trade studies?		

TABLE B-5. SOFTWARE DETAILED DESIGN CHECKLIST

Item	Yes/No	Comments
CLARITY		
1. Is the unit design—including data flow, control flow, and interfaces, clearly represented?		
COMPLETENESS		
1. Are all variables, pointers, and constants defined and initialized?		
COMPLIANCE		
1. Does documentation follow project and/or DoD standards?		
2. Has unit design been created using required methodology and tools?		
CONSISTENCY		
1. Are data elements named and used consistently throughout the unit and unit interfaces?		
2. Are designs of all interfaces consistent with each other and with preliminary design documents?		
CORRECTNESS		
1. Is there logic missing?		
DATA USAGE		
1. Are all declared data blocks actually used?		
FUNCTIONALITY		
1. Does this design implement the specified algorithm?		
2. Will this design fulfill its specified requirement and purpose?		

Item	Yes/No	Comments
INTERFACE		
1. Are all inputs and outputs properly defined and checked?		
2. Have parameters been specified in units of measure, range of values, accuracy, and precision?		
LEVEL OF DETAIL		
1. Are all required module attributes defined?		
2. Has sufficient detail been included to develop and maintain the code?		
MAINTAINABILITY		
1. Has complexity of this design been minimized?		
2. Do units exhibit clarity/readability/modifiability to meet maintenance requirements?		
PERFORMANCE		
1. Do processes have time windows?		
2. Have all constraints, such as processing time and size, been specified for each unit?		
RELIABILITY		
1. Is error checking performed on inputs, outputs, interfaces, and results?		
2. Are meaningful messages issued for all error conditions?		
TESTABILITY		
1. Can each unit be tested, demonstrated, analyzed, or inspected to show they satisfy requirements?		

Item	Yes/No	Comments
TRACEABILITY		
1. Are all parts of design traced back to the requirements?		
2. Can all design decisions be traced back to trade studies?		

TABLE B-6. FORTRAN SOURCE CODE CHECKLIST

Item	Yes/No	Comments
FUNCTIONALITY		
1. Do modules meet design requirements?		
2. Does each module have a single purpose?		
3. Is there some code in the module that should be a function or a subroutine?		
4. Are utility modules used correctly?		
5. Does the code match the design specifications?		
6. Does code impair performance of module (or program) to any significant degree?		
DATA USAGE		
A. General		
1. Are data defined?		
2. Are there undefined or unused variables?		
3. Are there typos, particularly the letter "O" for zero, and the letter "I" for one?		
4. Are there misspelled names that are compiled as function or subroutine references?		
5. Are declarations in the correct sequence? (DIMENSION, EQUIVALENCE, DATA).		
B. Common/Equivalence		
1. Are there local variables that are, in fact, misspellings of a COMMON element?		

Item	Yes/No	Comments
2. Are the elements in the COMMON in the right sequence?		
3. Do EQUIVALENCE statements force any unintended shared data storage?		
4. Is each EQUIVALENCE commented?		
C. Arrays		
1. Are all arrays Dimensioned?		
2. Are array subscript references in column, row order? (Check all indices in multi-dimensioned arrays.)		
3. Are array subscript references within the bounds of the array?		
4. Are array subscript references checked in critical cases?		
5. Is each array used for only one purpose?		
D. Variables		
1. Are variables initialized in DATA statements, BLOCK DATA, or previously defined by assignments or COMMON usage?		
2. Should variables initialized in DATA statements actually be initialized by an assignment statement?		
3. Are variables used for only one purpose?		
4. Are variables used for logical unit assignments?		
5. Are the correct types (REAL, INTEGER, LOGICAL, COMPLEX) used?		
E. Input and Output		
1. Do Formats correspond with the READ and WRITE lists?		
2. Is intended conversion of data specified in the FORMAT?		

Item	Yes/No	Comments
3. Are there redundant or unused FORMAT statements?		
4. Should this module be doing any I/O? Should it be using a message facility?		
5. Are messages understandable?		
6. Are messages phrased with the correct grammar?		
7. Does each line of a message fit on all of the expected output devices?		
F. Data		
1. Are all logical unit numbers and flags assigned correctly?		
2. Is the DATA statement used and not the PARAMETER statement?		
3. Are constant values constant?		
CONTROL		
A. Loops		
1. Are loop parameters expressed as variables?		
2. Is the initial parameter tested before the loop?		
3. Is the loop index within range of any array it is subscripting?		
4. Is the index variable only used within the DO loop?		
5. If the value of an index variable is required outside the loop, is it stored in another location?		
6. Does loop handle all conditions required? Error conditions?		
7. Does loop handle cases which may "fall through"?		
8. Is loop nesting in the correct order and can loops be combined?		
9. If possible, do nested loops process arrays as they are stored?		

Item	Yes/No	Comments
B. <u>Branches</u>		
1. Are branches handled correctly and commented?		
2. When using computed GO TOs, is the fall-through case tested, checked, and handled correctly?		
3. Are floating point comparisons done with tolerances and never made to an exact value?		
<u>LINKAGE</u>		
1. Does CALLing program have the same number of parameters as each routine?		
2. Are passed parameters in correct order and correct type?		
3. Are constant values passed via a symbol (variable) rather than being passed directly?		
4. Is an unused parameter named DUMMY, or some name which reflects its inactive status?		
5. Is an array passed to a subroutine only when an array is defined in the subroutine?		
6. Are the input parameters listed before the output parameters?		
7. Does the subroutine return an error status output parameter?		
8. Do return codes follow conventions?		
9. Are arrays used as intended?		
10. If array dimensions are passed (dynamic dimensioning) are they greater than 0?		
11. If a subroutine modifies an array, are the indices checked, or are the dimensions passed as parameters?		

Item	Yes/No	Comments
12. Does a subroutine modify any input parameter and end with RETURN statement?		
13. Does a FUNCTION routine have only one output value?		
COMPUTATION		
1. Are arithmetic expressions evaluated as specified?		
2. Are parentheses used correctly?		
3. Is use of mixed-mode expressions avoided?		
4. Are intermediate results stored instead of recomputed?		
5. Is all integer arithmetic involving multiplication and division performed correctly?		
6. Do integer comparisons account for truncation?		
7. Are complex numbers used correctly?		
8. Is precision length selected adequate?		
9. Is arithmetic performed efficiently?		
10. Can a multiplication be used instead of a division, if so, is it commented so as not to obscure process?		
MAINTAINABILITY		
1. Is non-standard FORTRAN isolated in subroutines and well documented?		
2. Is use of EQUIVALENCE limited so that it does not impede understanding the module?		
3. Is use of GO TOs limited so that it does not impede understanding the module?		

Item	Yes/No	Comments
4. Is there no self-modifying code? (No ASSIGN statements, or PARAMETER statements.)		
5. Is the module independent of specific devices where possible?		
6. Where possible, are the CALLing routine parameter names the same as the subroutine parameter names?		
7. Are type declarations implicit rather than explicit when possible?		
CLARITY		
1. Is the module header informative and complete?		
2. Are there sufficient comments to understand the code and are they informative?		

TABLE B-7. C SOURCE CODE CHECKLIST

Item	Yes/No	Comments
FUNCTIONALITY		
1. Does each module have a single function?		
2. Is there code that should be in a separate function?		
3. Is code consistent with performance requirements?		
4. Does code match design specifications?		
DATA USAGE		
A. Data and Variables		
1. Are all variable names in lower case characters?		
2. Are names of all internals distinct in 8 characters?		
3. Are names of all externals distinct in 6 characters?		
4. Do all initializers use "="? (v.7 and later); in all cases should be consistent.		
5. Are declarations grouped into externals and internals?		
6. Do all but the most obvious declarations have comments?		
7. Is each name used for a single function except single character variables ("c", "i", "j", "k", "n", "p", "q", "s")?		
B. Constants		
1. Are all constant names in upper case letters?		
2. Are constants defined via "# define"?		
3. Are constants used in multiple files defined in an INCLUDE header file?		

Item	Yes/No	Comments
C. Pointers Typing		
1. Are pointers declared and used as pointers (not integers)?		
2. Are pointers not typecast (except assignment of NULL)?		
CONTROL		
1. Are "else_if" and "switch" used clearly?		
2. Are "goto" and "labels" used only when absolutely necessary, and always with well-commented code?		
3. Is "while" rather than "do-while" used wherever possible?		
LINKAGE		
1. Are "INCLUDE" files used according to project standards?		
2. Are nested "INCLUDE" files avoided?		
3. Is all data local in scope (internal static/external static) unless global linkage is specifically necessary and commented?		
4. Are names of macros all upper case?		
COMPUTATION		
A. Lexical Rules for Operators		
1. Are unwary operators adjacent to their operands?		
2. Do primary operators "->" "." "()" have a blank space around them? (Should have none).		
3. Do assignment and conditional operators always have a blank space around them?		
4. Are commas and semicolons followed by a blank space?		

Item	Yes/No	Comments
5. Are keywords followed by a blank space?		
6. Is the use of "(" following a function name adjacent to the identifier?		
7. Are blank spaces used to show precedence?		
B. Evaluation Order		
1. Are parentheses used properly for precedence?		
2. Does code depend on evaluation order? Exceptions are as follows:		
a. expr1, expr2		
b. expr1? expire : expr3		
c. expr1 && expire		
d. expr1 || expire		
3. Are shifts used properly?		
4. Does code depend on order of effects? (e.g., i = i++;)?		
MAINTAINABILITY		
1. Are library routines used?		
2. Are non-standard usages isolated in subroutines and well documented?		
3. Does each module have one exit point?		
4. Is the module easy to change?		
5. Is the module independent of specific devices where possible?		
6. Is the system standard defined types header used if possible (otherwise use project standard header, by "include")?		
7. Is use of "int" avoided (use standard defined type instead)?		

Item	Yes/No	Comments
CLARITY		
A. Comments		
1. Is the module header informative and complete?		
2. Are there sufficient comments to understand the code?		
3. Are the comments in the modules informative?		
4. Are comment lines used to group logically-related statements?		
5. Are the functions of arrays and variables described?		
6. Are changes made to a module noted in the development history section of the header after its release?		
B. Layout		
1. Is the layout of the code such that the logic is apparent?		
2. Are loops indented and visually separated from the surrounding code?		
C. Lexical Control Structures		

1. Is a standard project-wide (or at lease consistent) lexical control structure pattern used?
For example:
`while (expr)`
`{`
`  stmts;`
`}`
or
`while (expr) {`
`stmts;`		

Item	Yes/No	Comments
}		

TABLE B-8. ADA SOURCE CODE CHECKLIST

Item	Yes/No	Comments
GENERAL		
1. Does the code implement the intent of preliminary/detailed design as documented in the design documents?		
2. Does implementation of the CSU adhere to standards and methodologies specified by the project?		
3. Is size of CSU within limits specified by project?		
4. Does code contradict information contained in any supporting software design documentation?		
5. Is there any logic error in the code?		
6. Is CSU program design internally consistent?		
7. Does CSU contain redundant code or excessively confusing/complex code?		
8. Does CSU perform a cohesive set of actions?		
9. Is CSU making use of available common software units (i.e., not duplicating code available for reuse)?		
DECLARATION AND TYPES		
1. Are all objects that do not change declared as constants?		
2. Does code use numeric literals or expressions in place of constant objects?		
3. Are constant objects declared without a type?		
4. Are separate or derived types being used for values that belong to logically independent sets?		

Item	Yes/No	Comments
5. Is an integer type used where an enumerated type is more appropriate?		
6. Are arrays/discrete types/groups of variables being used where record types would be more appropriate?		
7. Is an array object declared with an anonymous type (i.e., declared without a type identifier?)		
8. Are type/object declarations using meaningful identifiers and accompanied with explanatory comments?		
9. Does the program text use a consistent and clear indentation scheme?		
EXCEPTIONS AND ERROR HANDLING		
1. Are exceptions being used for normal processing, such as returning normal status information?		
2. Are exceptions being logged as specified by the project and programming guidelines in the ASG?		
3. Do task bodies provide a handler for all anticipated exceptions?		
4. Is there a "when others" handler in the outer most frame of each task body and the main program?		
5. Is there provision for explicit handling of all anticipated I/O exceptions?		
6. Do units interfacing with non-ADA environments transform error-status info into user defined exceptions?		
7. Is there a possibility of premature task termination due to unhandled or partially handled exceptions?		
8. Is any exception propagated outside its static scope?		
9. Are any checks suppressed using PRAGMA SUPPRESS?		

Item	Yes/No	Comments
10. Are the error messages produced clear, meaningful, and user-friendly?		
11. Does the CSU design leave the possibility of unhandled exceptions occurring during elaboration?		
12. Are any predefined exceptions being raised explicitly (e.g. constraint error)?		
GENERIC UNITS		
1. Does use of generic units conform to guidelines given by the project?		
2. With generics having subprogram parameters, are the actual subprograms provided during instantiation conceptually consistent with the corresponding generic formal parameters?		
3. Does prologue associated with the generic units conform to the guidelines given?		
4. Are generic instantiations properly commented?		
5. Does the CSU design make proper use of the ADA generic facilities? Are there repetitive portions of code that should be coded as generic units?		
INPUT/OUTPUT		
1. Is there any unnecessary use of Low_Level_IO?		
2. Are opened files being closed through every possible exit path?		
3. Does CSU validate all untrustworthy input (e.g. by applying range checks, etc.)?		
4. Does CSU provide a "user-friendly" interface for human computer interaction?		
LEXICAL ELEMENTS		

Item	Yes/No	Comments
1. Does code redefine meaning of any identifier:		
a. In the packages STANDARD, system, calendar or the predefined IO package?		
b. denoting an attribute of the entities declared in the STANDARD package?		
2. Do comments add information rather than merely paraphrase code?		
3. Does code deviate from the following lexical conventions:		
a. Reserved identifiers in lower case?		
b. Type identifiers in upper case?		
c. Identifiers are meaningful?		
d. Consistent indentation and layout or program text?		
4. Is there a standard prologue included with each package, subprogram, and task?		
NAMES AND EXPRESSIONS		
1. Are assignments to arrays/records being made using assignments to individual components vice aggregates?		
2. Are explicit type conversions used where a type qualified expression is meant?		
3. Is type qualification being used when it can be avoided?		
4. Are type names common nouns, such as DEVICE_TYPE, AUTHORITY_LEVEL_TYPE, USER_NAME_TYPE?		
5. Are names of non-Boolean objects expressed as nouns?		
6. Are names of the Boolean-valued objects expressed as predicate		

Item	Yes/No	Comments
clauses?		
PACKAGES		
1. Does each package fulfill one and only one of the following:		
a. Models an abstract entity appropriate to the problem domain?		
b. Collects a cohesive set of types and objects?		
c. Groups together program units for configuration control or visibility reasons?		
d. Implements a state machine?		
2. Are contents of package cohesive?		
3. Does logical hierarchy of packages and procedures reflect levels of abstraction?		
4. Are all non-trivial nested package bodies declared as subunits?		
5. Does private part of any package specification contain extraneous information?		
6. Is package name a meaningful noun phrase?		
7. Do package names have the prefix and suffix as specified?		
8. Does package specification prologue text conform to project standards?		
9. Do package body and body stub prologue texts conform to project standards?		
PROGRAM STRUCTURE AND COMPILATION		
1. Are all non-trivial nested units made into separate subunits?		
2. Does any ADA unit import another unit it does not need to see?		

Item	Yes/No	Comments
3. Do the CSU structure (each package a CSU), size, and contents conform to project guidelines?		
REPRESENTATION CLAUSES AND IMPLEMENTATION DEPENDENT FEATURES		
1. Are representation clauses and implementation-dependent features hidden inside package bodies?		
2. Are there comments to identify and document use of machine dependent (hardware/operating system), implementation-dependent (compiler) or ADA low-level features (representation clauses and other features)?		
3. Are representation clauses or implementation-dependent features being used except for following purposes:		
a. Increase efficiency to meet requirements?		
b. Interfacing hardware, foreign code, or foreign data?		
c. Interrupt handling?		
d. Specify task storage size?		
4. Are representation clauses placed close to objects they affect?		
5. Are representation clauses being used to change meaning of the program?		
STATEMENTS		
1. Are loops rather than array slice assignments being used to copy all or part of an array?		
2. Are "if" and "case" statements being used improperly?		
3. Are blocks being used in place of procedures?		

Item	Yes/No	Comments
4. Are there any "go to" statements?		
5. Are the related sequences of statements collected together into groups by blocking them with blank lines?		
6. Do the "if", "case", "loop", and block statements follow a consistent and meaningful indentation scheme?		
7. Is code well documented? Do comments accurately reflect logic of statements?		
SUBPROGRAMS		
1. Does each subprogram perform a single, conceptual action at its level of abstraction?		
2. Are overloaded functions being used in cases other than the following:		
a. Subprograms performing similar actions on different types of arguments		
b. Overloading of operators?		
3. Are procedure names expressed as imperative verbs?		
4. Are names of BOOLEAN-valued functions expressed as predicate clauses?		
5. Does subprogram specification prologue text conform to project guidelines?		
6. Are parameter modes missing from procedure specifications and are they correct?		
7. Is "in out" mode being used in cases where the "in" or the "out" mode is more appropriate?		
8. Do subprogram body and body stub prologue texts conform to project		

Item	Yes/No	Comments
guidelines?		
TASKS		
1. Are task types being used where a "directly names" task would be more appropriate?		
2. Is there unnecessary use of dynamically created tasks?		
3. Is there a proper termination mechanism for each task?		
4. Does the body of the accept statement contain any action(s) not essential for rendezvous?		
5. Does task directly or indirectly call its own entry?		
6. Does any task use a "busy wait" loop in place of a delay statement?		
7. Does code rely upon execution pattern of tasks for synchronization?		
9. Are there any concurrently executing tasks that share an unprotected common variable?		
10. Is each variable shared by tasks identified by documentary comments at its point of declaration?		
11. Is each task name a noun phrase describing the function of the task?		
12. Does task specification prologue text conform to project standards?		
13. Do task body and body stub prologue texts conform to project standards?		
14. Is each accept statement accompanied by the comments as specified by project guidelines?		
15. Is mode of any rendezvous parameter missing from the entry or accept statement?		

Item	Yes/No	Comments
16. Does the task rely upon its priority for synchronization with others?		
17. Is there a possibility of blocking a higher priority task because of "priority inversion"?		
VISIBILITY		
1. Does scope of any identifier (local or imported from another unit) extend further than necessary?		
2. Is a "use" clause included in cases other than the following:		
a. Text_IO, and instantiation of its components?		
b. To make overloaded operators from imported packages directly visible?		
3. Are all imported entities referenced with their full names?		
4. Are all data passed to subprograms parametric (i.e., subprograms do not receive or pass information via global variables)?		

TABLE B-9. TEST PLAN CHECKLIST

Item	Yes/No	Comments
COMPLETENESS		
1. Does test plan specify overall approach and policy for acceptance tests?		
2. Does test plan clearly specify order of steps of all integration testing?		
3. Does test plan include description of type of hardware and software system environment to be used?		
4. Does test plan define success (pass/fail) criteria for all tests?		
5. Does test plan adequately describe functions being tested?		
6. Does test plan describe conditions under which testing will be halted and resumed during integration test?		
7. Does test case set adequately exercise all significant code changes, particularly interface modifications?		
8. Does test plan adequately describe integration test baselines?		
9. Does test plan define sufficient and proper regression testing?		
COMPLIANCE		
1. Does test plan list all specifications, standards, and documents necessary for its development?		
CONSISTENCY		
1. Has the order of integration tests been defined to match the order specified in higher level documents?		
2. Is test plan consistent with higher level test plan documents?		
CORRECTNESS		

Item	Yes/No	Comments
1. Are test plan entrance and exit criteria realistic?		
2. Are all necessary drivers and stubs identified and available to test functions as specified?		
LEVEL OF DETAIL		
1. Does test case set include adequate coverage of illegal and conflicting input combinations?		
2. Does test case set include adequate usage of default input values?		
3. Does test case set exercise an adequate number of program error paths?		
MAINTAINABILITY		
1. Are changes to specifications, design, or coding that may occur during testing contained in test plan?		
RELIABILITY		
1. Is sufficient test data collected and documented to support estimation of software's reliability?		
TESTABILITY		
1. For all requirements considered untestable, are explanations provided as to why they are untestable?		
2. Has development/procurement of test facilities, methods and tools been scheduled with adequate lead time?		
3. Are testing schedules described in a sufficient level of detail?		
4. Is the method of estimating resource usage required for testing identified?		

Item	Yes/No	Comments
5. For multiple builds, have all requirements been identified on a per-build basis?		
6. Have roles and responsibilities for all personnel involved in test activity been identified?		
7. Is specification of test facilities consistent with test success criteria?		
8. Are there any scheduling conflicts among testing personnel schedules?		
9. Does test plan call for participation of independent quality assurance personnel to verify test activity?		
TRACEABILITY		
1. Do acceptance tests exercise each requirement specified in higher level documents?		
2. Are test acceptance criteria traceable to higher-level requirements documents?		
3. Does the test case set for integration testing exercise each interface described in higher level documents?		

TABLE B-10. TEST CASES AND PROCEDURES CHECKLIST

Item	Yes/No	Comments
CLARITY		
1. Are user instructions explicit and clear for ease of execution of test procedure?		
2. Are user instructions presented step-by-step and in order they must be performed?		
3. Are steps of set-up and test procedures precise, unambiguous, and listed as individual items?		
4. Are there "progress" messages that will notify user when significant parts of test are executed?		
5. Are criteria for success/failure clear and unambiguous?		
COMPLETENESS		
1. Is function being tested accurately described?		
2. Is function being tested the latest revision?		
3. Is description of purpose of this test procedure complete and accurate?		
4. Is each requirement associated with this function exercised by this test procedure?		
5. Is expected response to each step of test procedure described with user instructions for that step?		
6. Does test procedure list precedence of tests?		
7. Does test procedure indicate significance of proper evaluation of test results?		

Item	Yes/No	Comments
8. Does procedure state whether or not it is possible to continue in event of a program stop or indicated error?		
9. Are an adequate number of control paths in a tested function exercised?		
10. Do test procedures lead to determination of success or failure?		
11. Are an adequate number of logical condition expressions in tested function exercised?		
12. Do test cases demonstrate program's response to illegal and conflicting input data?		
CONSISTENCY		
1. Are all dependencies of test procedure identified?		
CORRECTNESS		
1. Do observed results of performing procedure agree with expected program behavior?		
2. Are interfaces between the code being tested and test equipment and software correct?		
PERFORMANCE		
1. If performance criterion is associated with any step of test procedure, is it explicitly stated?		
RELIABILITY		
1. Has test equipment been validated and calibrated?		
2. Has test software been validated?		
3. Have all input data been verified?		

Item	Yes/No	Comments
4. Is sufficient test data collected and documented to support estimation of software's reliability?		
TESTABILITY		
1. Does test procedure identify all equipment, software, and personnel required for testing?		
2. Can test procedure be performed with minimal support from development team?		
3. Is test procedure consistent with the capabilities of test facilities?		
TRACEABILITY		
1. Does test procedure list all specifications, procedures, handbooks, or manuals required for operation?		
2. Is traceability shown between requirements and acceptance test combinations?		

TABLE B-11. SOFTWARE USER DOCUMENTATION CHECKLIST

Item	Yes/No	Comments
CLARITY		
1. Is terminology consistent with users' operational understanding of terms?		
2. Is documentation easy-to-read and easy-to-understand for end-users?		
3. Is software user's guide written at the level of understanding for users?		
COMPLETENESS		
1. Is user's guide complete in terms of discussion of all available functionality?		
2. Is user's guide complete with respect to other pertinent references for users?		
3. Is user's guide complete and cross-referenced to functional requirements?		
COMPLIANCE		
1. Is user's guide in compliance with approved reference(s) for documentation?		
2. Is user's guide in compliance with commercially-acceptable documentation?		
DATA USAGE		
1. Is data usage within user's guide tied to databases documented in the design?		
2. Is data usage within user's guide dependent on other databases using design?		

FUNCTIONALITY		
1. Are requirements for on-line assistance documented within user's guide?		
2. Are requirements for on-line tutorial documented within user's guide?		
INTERFACE		
1. Are interfaces with other software documented for reference in user's guide?		
2. Are graphical user interface applications documented within user's guide?		
LEVEL OF DETAIL		
1. Is level of detail tailorable to meet diverse needs from novice to expert users?		
2. Is level of detail sufficiently documented in user's guide to support novices?		
MAINTAINABILITY		
1. Is documentation for user's guide in format designed to aid maintainability?		
2. Is documentation referenced to software version releases to be up-to-date?		
PERFORMANCE		
1. Have performance standards needed by end-users been cited in user's guide?		
2. Have limits and defaults on performance been documented in user's guide?		

APPENDIX C. FORMS AND CHECKLISTS

This appendix provides all required forms and checklists needed to plan and conduct peer reviews, as well as instructions for their use, completion and distribution.

Peer Review Announcement and Report (Form 1). This form is used to record peer review measurement data for Walkthroughs, Technical Reviews, and Formal Inspections (FI). Refer to Section C.1 in this appendix for the form and preparation instructions.

Peer Review Defect Log (Form 2). This form is used to record defects discovered during FIs. It may also be used for Walkthroughs and Technical Reviews, if desired. Refer to Section C.2 in this appendix for the form and preparation instructions.

Inspection Checklists. The inspection checklists are recommended to assist participants in preparing for and performing their assigned roles during an FI. Refer to Section C.3 in this appendix for the checklists.

C.1 PEER REVIEW ANNOUNCEMENT AND REPORT (FORM 1)

The Peer Review Announcement and Report (Form 1) in Table C-1 is provided to announce an upcoming peer review and to assist in collecting and recording required peer review measurement data. The Peer Review Announcement and Report (Form 1) is available as a standalone form and can be downloaded from the SSC San Diego PAL at: http://sepo.spawar.navy.mil.

Usage: The Moderator or Review Leader prepares the first page of this form during the planning phase, while the Moderator or Review Leader prepares the second page upon completion of the rework phase.

C.1.1 Form 1 Announcement Portion Instructions

Instructions for completing selected fields in the "Announcement" portion of the form are provided below. Cross-references to other sections in the main body of this document are also provided for amplifying information and details on certain fields.

a. Project And Product Information:

   1. Project Name: This field identifies the project by name.
   2. Project Code: This field identifies the project by code.
   3. Work Product Name (Title and Section): This field identifies the work product being reviewed.
   4. Work Product Type: This field identifies the type of work product to be reviewed (document, source code, database, etc.).
   5. Size of Product: This field identifies the number of pages, lines of code, or other appropriate size measure to be reviewed in the work product.
   6. Review Type: This field identifies the type of review; WT for Walkthrough, TR for Technical Review, or FI for Formal Inspection.

b. Review Schedule:

   1. Announcement Date: This field contains the date on which the review was announced. It is suggested the announcement be made a minimum of three-working days prior to the scheduled review meeting date.

2. Overview Meeting Date: This field is optional, and if used contains the scheduled overview meeting date.
3. Review Meeting Date: This field contains the scheduled review meeting date. Schedule the review meeting at a mutually acceptable time, date and location. This meeting should be scheduled a minimum of three working days after the announcement of the review.
4. Defect Logs Due Date: This field contains the date that completed defect logs are to be returned to the moderator/review leader.
5. Actual Review Meeting Date: This field contains the date the review meeting was actually held.
6. Review Process Completion Date: This field contains the date the Peer Review Process was completed.

c. Peer Review Participants: This section lists the review participants and their assigned role in the review. The roles are defined in sections 2.5.1.2, 2.5.2.2, and 2.5.3.2 of the main document.

C.1.2 Form 1 Report Portion Instructions

Instructions for completing selected fields in the "Report" portion of the form are as follows:

a. Effort (In Hours): This section provides spaces for recording labor hours expended by the reviewers, moderator/review leader, and author(s)during the review phases. Note these data are entered to the nearest quarter of an hour (e.g., 1.75). Labor hours for the observer (optional) are combined with those of other reviewers, and are reported under the "Review Meeting" heading in the table.

b. Defect Summary By Type:

1. Type Code: Enter a defect type code from Table C-2
2. Status: Enter a status of Major, Minor, or Open for each type of defect discovered.
3. When Discovered: Report the number of recorded defects or issues (by type and status) discovered during preparation and at the review meeting.
4. Resolved/Reworked: Report the total number of defects and open issues by type corrected by the author or others when the product was reworked.
5. Totals: Record the totals for each column.

TABLE C-1. PEER REVIEW ANNOUNCEMENT AND REPORT (FORM 1)

PROJECT AND PRODUCT INFORMATION **REVIEW SCHEDULE**

Project Name		Announcement Date	
Project Code		Overview Meeting Date	
Work Product Name (Title and Sections)		Review Meeting Date	
Work Product Type		Defect Logs Due Date	
Size of Product (pages/SLOC/other)		Actual Review Meeting Date	
Review Type	*WT___ *TR___ *FI____	Review Process Completion Date	

Is this a 2nd Review? (Yes/No)
Is a 2nd Review Required? (Yes/No)

*WT = Walkthrough
*TR= Tech Review
*FI= Formal Inspection

PEER REVIEW PARTICIPANTS

Role	Name	Organization	Code	Phone

(Roles vary by type of Peer Review)

Total Number of Participants in Attendance at Review Meeting: _____

EFFORT (IN HOURS)

	Planning	Overview	Preparation	Review Meeting	Hrs to Date Subtotal	Rework	Follow-Up	Total
Reviewer(s)								
Author(s)								
Moderator/ Review Leader								
TOTALS								

DEFECT SUMMARY by Type (See Table C-2 for defect type definitions)

		When Discovered					When Discovered		
Type Code	Status*	Preparation	Review Meeting	Resolved/ Reworked	Type Code	Status*	Preparation	Review Meeting	Resolved/ Reworked
Totals					Totals				

*Status Entries: Major, Minor or Open

Moderator or Review Leader's Signature: _____

TABLE C-2. DEFECT TYPE CODE DEFINITIONS

Coding Defects	
Type	**Description**
C10 Documentation	Comments, messages that are missing or incorrect
C20 Syntax	Spelling, punctuation, typos, instruction formats
C30 Build, package	Change management, library, version control
C40 Assignment	Declaration, duplicate names, scope, limits
C50 Interface	Procedure calls and references, I/O, User formats
C60 Checking	Error messages, inadequate range checks, array bounds checks
C70 Data	Structure, content
C80 Function	Logic, pointers, loops, recursion, computation, function defects
C90 System	Configuration, timing, memory
C100 Environment	Design, compile, test, other support system problems
Document Defects	
D10 Format (Redline type errors)	
D20 Missing/Incorrect data	
D30 Misplaced data	
D40 Missing reference	
D50 Incorrect reference	
D60 Un-referenced figure or table	
D70 Inconsistent (doesn't follow standard)	
D80 Requirement Error	
D90 Design Error	
D100 Interface Error	

C.2 PEER REVIEW DEFECT LOG (FORM 2)

The form in Table C-3 is used (1) during the Preparation Phase of an FI for recording individual defects by FI inspectors; (2) during the Inspection meeting for preparing a collated, integrated defect log, and (3) by the author during the Rework Phase. This form is included as part of the FI package. It may also be used during a walkthrough or technical review.

C.2.1 Form 2 First Page Instructions

The following instructions are provided for completing selected portions of the defect log:

a. Preparation (Date, Hours and Total Hours): This section provides spaces for recording the actual date(s), corresponding hours, and total hours spent in reviewing the work product for major and minor defects and "redline" errors.

b. Focus Area #1/#2: This section lists an area(s) assigned to reviewers for particular emphasis/attention when reviewing the work product. Examples include: clarity, completeness, compliance, consistency, correctness/logic, data usage, functionality, interfaces, level of detail, maintainability, performance, reliability, testability, and traceability.

c. Return this form: It is recommended the moderator/review leader assign a due date for returning completed defect logs a minimum of one working day before conducting the review meeting.

d. Defects Summary: This section provides spaces for recording the number of major, minor and open issue defects found during the preparation and review/inspection meeting phases, and corresponding totals for each phase. Note: Reviewers need only provide numeric entries under the "Preparation" heading when completing this part of the defect logs. The recorder provides separate and total counts for all "approved" defects and open issues (i.e., those discussed and approved by group consensus during the review meeting).

e. Defects: The following guidance is provided to assist reviewers and the recorder in describing and classifying defects/open issues during the preparation and review meeting phases:

1. #—A consecutive number (beginning with 1) assigned to each defect detected in the work product. If the defect logs are to be merged prior to the meeting, this field should be left blank.
2. Init—two or three character initial for inspector's name.
3. Location—The specific area in the work product (e.g., line number, section, paragraph, page, line of source code) containing the defect summarized under the Description heading.
4. Description—A brief (e.g., one to three-line) summary of a detected defect or open issue; it should not propose a solution for the defect or method for resolving an open issue.
5. Defect Type — Code from list of defect types in Table C-2 that describes the defect.
6. Major—A significant error or omission in a work product that will result in a malfunction or unexpected outcome if uncorrected. Major defects are considered the highest priority problems in software development. Reviewers are asked to place a check mark in the column to the left of the dashed line when assigning defects to this category during the preparation phase; and in the column to the right of the dashed line once consensus is reached on this classification of a defect(s) during the review meeting.
7. Minor—An error or omission that does not cause or lead to a malfunction. Minor defects are considered low priority problems in software development. Use the same marking system described for Major defects when making entries to this field.
8. Open—(Review meeting only) A problem not easily classified as a major/minor defect, or any defect under discussion for more than two minutes during a review meeting which is subsequently deferred for resolution. Use the same marking system described for Major defects in this field.

9. Not— Defects submitted during the review meeting that are not considered (via group consensus) to be valid for corrective action or resolution.

C.2.2 Form 2 Continuation Sheet Instructions

Table C-4 is the continuation sheet for the Peer Review Defect Log. Review participants should make multiple copies of the continuation sheet prior to documenting and classifying defects in a work product.

Each continuation sheet should be numbered consecutively (e.g., Sheet 2 of 7) to ensure defects reported on each form can be easily "mapped" to relevant portions of the work product, and to minimize the possibility of forms being intermingled with those submitted by other participants. All other fields on the continuation sheet are either self-explanatory or use the same guidance as provided in Section C.2.1.

This page intentionally left blank

TABLE C-3. PEER REVIEW DEFECT LOG (FORM 2)

Sheet _____ of _____

Project	Reviewer Name	Code	Phone	Product Type	Preparation	
					Date	Hours
Subsystem	Role in Review Process: __ Moderator/Rvw Leader	Focus Area #1		Date Received		
Unit(s)	__ Presenter __ Reviewer __Recorder	Focus Area #2		Date Completed	Total Hours	

Return this form to the review leader at least 8 hours before the scheduled review meeting	Review Type: WT__ TR__ FI__	Yes	No
	I am prepared for my role in the review.	___	___
	I think this work product is ready for review.	___	___
	2nd review after rework is necessary (to be decided at the end of the review meeting.)	___	___

Defects Summary		
	Preparation	Review Mtg
Major		
Minor		
Open		
Total		

Defects

#	Init	Location(s)	Description	Defect Type*	Major	Minor	Open	Not

*** See back of Form 2 for Defect Type Codes**

Coding Defects	
Type	**Description**
C10 Documentation	Comments, messages that are missing or incorrect
C20 Syntax	Spelling, punctuation, typos, instruction formats
C30 Build, package	Change management, library, version control
C40 Assignment	Declaration, duplicate names, scope, limits
C50 Interface	Procedure calls and references, I/O, User formats
C60 Checking	Error messages, inadequate range checks, array bounds checks
C70 Data	Structure, content
C80 Function	Logic, pointers, loops, recursion, computation, function defects
C90 System	Configuration, timing, memory
C100 Environment	Design, compile, test, other support system problems
Document Defects	
D10 Format (Redline type errors)	
D20 Missing/Incorrect data	
D30 Misplaced data	
D40 Missing reference	
D50 Incorrect reference	
D60 Un-referenced figure or table	
D70 Inconsistent (doesn't follow standard)	
D80 Requirement Error	
D90 Design Error	
D100 Interface Error	

TABLE C-4. PEER REVIEW DEFECT LOG (FORM 2) (CONTINUATION SHEET)

Sheet _____ of _____

Project	Inspector Name	Code	Phone	Date Received
Subsystem	Unit(s)	Review Type: WT__ TR__ FI__		

Defects

#	Init	Location(s)	Description	Defect Type*	Major	Minor	Open	Not

*** See Table C-2 for Defect Type Codes**

C.3 INSPECTION CHECKLISTS

The checklists in this section are intended for reproduction and distribution to inspection team members. Note the items in each checklist (1) outline responsibilities of team members during various phases of the FI Procedure, and (2) may be tailored to individual project needs and the type of work product to be inspected.

INSPECTOR'S CHECKLIST

Date/Initials

Planning:

- Confirm availability to serve in assigned role during the FI by consulting with project manager or supervisor as applicable. ____/_____
- Record time spent in Planning Phase: _____ ____/_____

Overview:

- Attend overview meeting; confirm assigned role and/or multiple roles (as applicable) for the inspection. ____/_____
- Record time spent in Overview Phase: _____ ____/_____

Preparation:

- Confirm receipt of the FI package and any other information provided by the moderator ____/_____
- Review pertinent checklists, guidelines, and standards cited in the FI package. ____/_____
- Examine work product as instructed by the moderator, emphasizing assigned focus areas, if applicable. ____/_____
- Classify and document all suspected defects (major/minor) on the Peer Review Defect Log (Form 2). ____/_____
- Record time spent in Preparation Phase: _____ ____/_____

Inspection:

- Introduce defects/issues pertinent to the area being paraphrased by the presenter. ____/_____
- Work with other inspectors during inspection to reach consensus on defects, their classification (i.e., major or minor), and open issues. ____/_____
- Provide feedback and lessons learned as requested by the moderator upon completion of the FI meeting. ____/_____
- Provide marked-up copy of work product ("redline" errors) to author at inspection meeting. ____/_____
- Record time spent in Inspection Phase: _____ ____/_____

Follow-up:

- Submit this checklist and other appropriate data to moderator. ____/_____

MODERATOR'S CHECKLIST

Date/Initials

Planning:

- Work with project management and author to select participants and assign roles. ____/_____
- Determine the size of the work product to be inspected. ____/_____
- Determine the need for overview meeting and schedule accordingly. ____/_____
- Assemble and distribute the FI package. ____/_____
- Confirm receipt of the FI packages by inspection team members. ____/_____
- Coordinate with presenter, recorder, author, and observer on their roles during the inspection meeting. ____/_____
- Record time spent in Planning Phase: _____. ____/_____

Overview:

- Confirm product understanding; explain assigned roles and focus areas; address any questions and concerns of team members. ____/_____
- Record time spent in Overview Phase: _____. ____/_____

Preparation:

- Prepare for inspection using checklist for inspectors. ____/_____
- Review completed defect logs for correct documentation. ____/_____
- Determine whether team members have prepared adequately for the inspection. ____/_____
- Confirm readiness of work product for inspection meeting. ____/_____
- Schedule inspection meeting. ____/_____
- Record time spent in Preparation Phase: _____. ____/_____

Inspection:

- Review roles, focus areas, and guidelines to be used for inspection. ____/_____
- At conclusion of two hour inspection meeting, decide if additional time (third hour) is required. ____/_____
- At conclusion, assign open issues and due dates for resolution. ____/_____
- At conclusion, decide if re-inspection of the product is needed. ____/_____
- At conclusion, solicit feedback from team members and observer. ____/_____
- At conclusion, consult with author regarding estimated rework. ____/_____
- Record time spent in Inspection Phase: _____. ____/_____

Follow-up:

- Verify defects/"redline" errors have been corrected and confirm all open issues have been resolved. ____/_____
- Collect checklists from inspection team members. Complete report portion of Form 1 and distribute FI results and metrics. ____/_____

AUTHOR'S CHECKLIST

Date/Initials

Planning:

- Confirm readiness of completed work product for FI. ____/_____
- Recommend selection of inspection participants to moderator. ____/_____
- Assist moderator in determining the size of the work product to be inspected. ____/_____
- Assist moderator in determining the need for an overview meeting. ____/_____
- Record time spent in Planning Phase: _____ ____/_____

Overview:

- Prepare briefing on the work product for an overview meeting (if required). ____/_____
- Address concerns or issues raised by participants about product. ____/_____
- Record time spent in Overview Phase: _____ ____/_____

Inspection:

- Acknowledge reported defects and open issues to work product. ____/_____
- Give brief technical explanations as required about work product. ____/_____
- Record time spent in Inspection Phase: _____ ____/_____

Rework:

- Provide rescheduling information for defects or open issues requiring more time for correction or resolution. ____/_____
- Correct all defects (major/minor) documented during the inspection. ____/_____
- Resolve all assigned open issues documented during the inspection. ____/_____
- Correct all "redline" errors documented as part of the inspection. ____/_____
- Record time spent in Rework Phase: _____ ____/_____

Follow-up:

- Verify all defects have been corrected during rework phase. ____/_____
- Verify assigned open issues have been resolved during rework. ____/_____
- Cooperate with moderator if further corrective action is needed. ____/_____
- Cooperate with moderator/inspectors if a re-inspection is necessary. ____/_____
- Record time spent in Follow-up Phase: _____ ____/_____
- Submit this checklist and other appropriate data to moderator. ____/_____

PRESENTER'S CHECKLIST

Planning:	**Date/Initials**
• Coordinate with the moderator concerning the best way to keep the inspection focused and effectively paced during the inspection meeting, with the goal of completing the review of the work product within a two hour period.	____/_____
• Record time spent in Planning Phase: _____	____/_____
Overview:	
• Attend and participate in the overview meeting (if required).	____/_____
• Record time spent in Overview Phase: _____	____/_____
Preparation:	
• Review pertinent checklists, guidelines, and standards provided with the FI package to support the presenter's additional role as an inspector.	____/_____
• Examine the work product with emphasis on assigned focus area(s) if this responsibility is performed in addition to assigned role as presenter during the FI.	____/_____
• Prepare for presentation during inspection meeting by annotating the work product to facilitate pacing and paraphrasing.	____/_____
• Practice paraphrasing the work product at a pace which allows for completion of the review within a two hour inspection period.	____/_____
• Record time spent in Preparation Phase: _____	____/_____
Inspection:	
• Following the moderator's opening remarks concerning the conduct of the inspection meeting, call for global defects/issues affecting the entire work product.	____/_____
• Guide participants through the work product by paraphrasing each section as it is inspected and set the pace of the meeting to maximize productivity—speeding up or slowing down as appropriate during discussions and when obtaining group consensus.	____/_____
• Encourage equitable participation through use of a "round-robin" method to rotate the discussion of defects and open issues among participants.	____/_____
• Participate as an inspector on an equal basis with others—balancing the dual roles of presenter and inspector.	____/_____
• Record time spent in Inspection Phase: _____	____/_____
Follow-up:	
• Submit this checklist and other appropriate data to moderator.	____/_____

RECORDER'S CHECKLIST

Planning:	**Date/Initials**
• Coordinate with the moderator concerning the best way to record defects, open issues, action items and their due dates, participant feedback/lessons learned.	____/_____
• Record time spent in Planning Phase: _____	____/_____
Overview:	
• Attend and participate in the overview meeting.	____/_____
• Record time spent in Overview Phase: _____	____/_____
Preparation:	
• Review pertinent checklists, guidelines, and standards provided with the FI package to support the recorder's additional role as an inspector.	____/_____
• Examine the work product with emphasis on assigned focus area(s) if this responsibility is performed in addition to assigned role as recorder during the FI.	____/_____
• Prepare for the role of recorder by reviewing plans, approach, and resources used to facilitate rapid recording of recorded data.	____/_____
• Record time spent in Preparation Phase: _____	____/_____
Inspection:	
• When group consensus is reached, record pertinent information on each defect or issue using clear and concise wording.	____/_____
• Advise inspection participants if the rate of providing verbal information about each defect/issue exceeds the recorder's ability to document them satisfactorily.	____/_____
• Once a defect/issue and its classification have been recorded, read it back aloud to the participants to ensure the information recorded in the defect log is both complete and accurate.	____/_____
• Participate as an inspector on an equal basis with others—balancing the dual roles of recorder and inspector.	____/_____
• Record action items, personnel assignments and due dates at the conclusion of the inspection meeting to facilitate tracking to completion.	____/_____
• Record feedback and lessons learned from the inspection based on the moderator's instructions for documenting this information.	____/_____
• Record time spent in Inspection Phase: _____	____/_____
Follow-up:	
• Submit this checklist and other appropriate data to moderator.	____/_____

OBSERVER'S CHECKLIST

Date/Initials

Planning:

- Coordinate with the moderator regarding the best way to capture feedback and record observations on a not-to-interfere basis with inspection plans. ____/_____
- Record time spent in Planning Phase: _____ ____/_____

Preparation:

- Review this checklist and tailor it to the FI meeting based on the type of work product to be inspected, together with guidance from the moderator. ____/_____
- Record time spent in Preparation Phase: _____ ____/_____

Inspection:

- Record observed strengths and areas needing improvement relative to individual and group roles; major differences between published guidance (checklists) and actual performance of duties; adherence to rules/protocols during the inspection meeting. ____/_____
- If requested by the moderator, brief inspection team members on methods for improving both individual and group performance and the FI Procedure itself. ____/_____
- Prepare a written summary of observations and submit to the moderator at the completion of the inspection meeting. ____/_____
- Record time spent in Inspection Phase: _____ ____/_____

Follow-up:

- Submit this checklist and other appropriate data to moderator. ____/_____

Template for Recording Observer's Comments

Observer's Name: _____ Date of Inspection:_____

Product Inspected: _____ Inspection Start Time:_____

_____ Inspection End Time: _____

Record observations about participant's effectiveness in performing duties:

Moderator: (name/code: _____/_____) _____

_____

_____

_____

Author: (name/code: _____/_____) _____

_____

_____

_____

Presenter: (name/code: _____/_____) _____

_____

_____

_____

Recorder: (name/code: _____/_____) _____

_____

_____

_____

Inspectors: names/codes: _____/_____ _____/____

_____/_____ _____/____

_____

_____

_____

_____

Document methods and rules used by participants in terms of enforcing protocols during the inspection meeting:

Record group process and synergy demonstrated and process efficiency noted:

Document strengths and areas in need of improvement for all participants:

DOCUMENT CHANGE REQUEST (DCR)

Document Title: **PEER REVIEW PROCESS**	Tracking Number:
Name of Submitting Organization:	
Organization Contact:	Phone:
Mailing Address:	
Short Title:	Date:
Change Location: (use section #, figure #, table #, etc.)	
Proposed change:	
Rational for Change:	

Note: For the Systems Engineering Process Office (SEPO) to take appropriate action on a change request, please provide a clear description of the recommended change along with supporting rationale.

Changes requests may also be submitted online via the SSC San Diego PAL http://sepo.spawar.navy.mil

Send to: Commanding Officer, Space and Naval Warfare Systems Center, 212, 53560 Hull Street, San Diego, CA 92152-5001 or
Fax to: (619)553-6249 or
Email to: sepo@spawar.navy.mil

DCR Form 7/2002